Employing Semantic Context for Sparse Information Extraction Assessment

PEIPEI LI, Hefei University of Technology
HAIXUN WANG, Amazon
HONGSONG LI, Alibaba Group
XINDONG WU, University of Louisiana at Lafayett

A huge amount of texts available on the World Wide Web presents an unprecedented opportunity for information extraction (IE). One important assumption in IE is that frequent extractions are more likely to be correct. Sparse IE is hence a challenging task because no matter how big a corpus is, there are extractions supported by only a small amount of evidence in the corpus. However, there is limited research on sparse IE, especially in the assessment of the validity of sparse IEs. Motivated by this, we introduce a lightweight, explicit semantic approach for assessing sparse IE.[1] We first use a large semantic network consisting of millions of concepts, entities, and attributes to explicitly model the context of any semantic relationship. Second, we learn from three semantic contexts using different base classifiers to select an optimal classification model for assessing sparse extractions. Finally, experiments show that as compared with several state-of-the-art approaches, our approach can significantly improve the *F*-score in the assessment of sparse extractions while maintaining the efficiency.

CCS Concepts: • **Information systems** → **Information retrieval**; *Evaluation of retrieval results*; • **Computing methodologies** → Machine learning approaches;

Additional Key Words and Phrases: Sparse information extraction, isA relationship, classification, semantic network

ACM Reference format:
Peipei Li, Haixun Wang, Hongsong Li, and Xindong Wu. 2018. Employing Semantic Context for Sparse Information Extraction Assessment. *ACM Trans. Knowl. Discov. Data.* 12, 5, Article 54 (June 2018), 36 pages.
https://doi.org/10.1145/3201407

[1]Source codes and data sets are available at below. https://github.com/peipeilihfut/AssessSparseIE.

A preliminary version of this paper was published in the Proceeding of 22nd ACM International Conference on Information and Knowledge Management(CIKM'13), pp. 1709-1714, 2013.
This work was supported in part by the National Key Research and Development Program of China under grant 2016YFB1000901, the Program for Changjiang Scholas and Innovative Research Team in University (PCSIRT) of the Ministry of Education under grant IRT17R32, the Natural Science Foundation of China under grants (61503112,61673152, 91746209), the US National Science Foundation under grant IIS-1652107, and the Natural Science Foundation of Anhui province under grant 1708085QF142.
Authors' addresses: P. Li, Hefei University of Technology, 193 Tunxi Rd, Hefei, Anhui Province 230009, China; email: peipeili@hfut.edu.cn; H. Wang, Amazon, Palo Alto, CA; email: haixun@gmail.com; H. Li, Alibaba Group, 969 West Rd, Hangzhou, Zhejiang Province, China; email: hongsong.lhs@alibaba-inc.com; X. Wu, University of Louisiana at Lafayett, 222 James R. Oliver Hall, Lafayett, LA; email: xwu@louisiana.edu.

https://doi.org/10.1145/3201407

1 INTRODUCTION

The explosive growth and popularity of the World Wide Web has resulted in a huge amount of texts on the Internet, which presents an unprecedented opportunity for information extraction (IE). IE is at the core of many emerging applications, such as entity search, text mining, and risk analysis using financial reports. In these applications, we can divide the outcome of IE into two categories according to the frequency: *heads* and *tails*. The *heads* are those that occur very frequently in the corpus. For instance, we can extract the fact that *google is a company* from numerous distinct sentences. It is built on the assumption that the higher the frequency, the more likely it is correct. Nevertheless, there are results that occur very infrequently, for instance, suppose from a corpus, we extract a statement that says *Rhodesia*[2] *is a country*, and its occurrences in the corpus are few and far between. In Table 1, we show some frequent and rare candidate countries extracted from a Web corpus using Hearst patterns (Hearst 1992). It turns out that all frequent entities are correct, while the majority of infrequent ones are incorrect. The mistakes come from either the extraction algorithm, or erroneous sentences in the corpus.

As we know, the distribution of words and phrases in a corpus of natural language utterances follows the Zipf's law[3] which states that the frequency of any word or phrase is inversely proportional to its rank in the frequency table, namely the long tail challenge (Wu et al. 2008). Therefore, it is a significant and challenging issue in IE to verify the correctness of an extraction in the long tail, also known as sparse extraction. This is because their occurrences in a particular syntactic pattern we use for extraction are very small. Thus, without a good mechanism to identify extractions correctly, sparse IE will suffer from either low precision or low recall.

1.1 State-of-the-art Approaches

Existing efforts in IE or sparse extraction can be divided into the following four classes.

Heuristic-based approaches such as Zhu et al. (2009), Zhang et al. (2015), and Amal et al. (2017), begin with a set of seed entities[4] given a relation or some prior label distributional knowledge, and they iteratively recognize extraction patterns for the relation. Nevertheless, erroneous entities will be produced due to doubtful extraction patterns caused by random extraction errors during the iteration. Redundancy-based approaches, such as Etzioni et al. (2005) and Downeya et al. (2010), require extractions to appear frequently with a limited set of patterns. However, these approaches assume that extractions drawn more frequently from distinct sentences in a corpus are more likely to be correct, and they are hence ineffective at assessing the correctness of sparse extractions because the extraction frequency follows a Zipf distribution. Knowledge-based approaches including (Schmitz et al. 2012) and Lin et al. (2016) identify IE in terms of external resources, such as Wikipedia,[5] WordNet (Ritter et al. 2009), and Freebase.[6] Thus, the coverage of class space in knowledge databases will limit the scalability of the aforementioned approaches.

In addition, context-based model building approaches use an important hypothesis known as the *distributional hypothesis* (Harris 1985), which says that different entities of the same semantic relation tend to appear in similar textual contexts. For example, we may not find many occurrences of *Rhodesia* in the Hearst pattern "countries such as Rhodesia." But if *Rhodesia* appears in a similar context where terms such as *India*, *USA*, and *Germany* occur, then we will be more certain about

[2]Rhodesia was an unrecognized state located in southern Africa that existed between 1965 and 1979 following its Unilateral Declaration of Independence from the United Kingdom on 11 November 1965.
[3]http://en.wikipedia.org/wiki/Zipf_law.
[4]Seed entities indicate some popular ones belonging to a given class or a given relation.
[5]Wikipedia Database: https://en.wikipedia.org/wiki/Wikipedia:Database_download.
[6]Freebase Data Dumps: https://developers.google.com/freebase/data.

Table 1. Frequent and Infrequent Candidate Entities of Country

Frequent entities	Rare entities
India	Northern
China	Sabah
Germany	Yap
Australia	Parts of Sudan
Japan	Wealthy
France	Western Romania
Canada	American artists
USA	South Korea, Japan
Brazil	New Sjaelland
Italy	Rhodesia

the claim that *Rhodesia* is a country according to the *distributional hypothesis*. This hypothesis is beneficial to assess sparse extractions, but the challenge lies in modeling contexts and measuring the semantic similarity of two sets of contexts.

In a naïve approach, a bag-of-words can be used to represent a context. This bag-of-words approach can easily be extended by using bigrams, trigrams, and the like, instead of unigram words, but this method is not semantic for context modeling and has a lower accuracy. Thus, more advanced approaches have been proposed, such as the REALM system (Ahuja and Downey 2010; Downey et al. 2007) based on a Hidden Markov Model (HMM) (Baum and Petrie 1966), an unsupervised learning algorithm called GloVe (Pennington et al. 2014) based on word–word co-occurrence using the tool of word2vec (Mikolov et al. 2013), and a deep learning approach for extracting manufacturing relationships (Leng and Jiang 2016). Nevertheless, aforementioned approaches present the following disadvantages. First, most approaches such as REALM and GloVe represent the feature space of texts using the distribution of a set of hidden states or a set of words instead of entities, which may lead to a worse performance in the assessment of sparse extractions. This is because each entity in the text can be an arbitrary multi-word expression instead of just a single word. For example, contexts for entities such as "new," "new york," and "new york times" are very different. Second, it is time-consuming to train the context-based models such as the HMM model and the deep learning model. For example, the time required to learn the parameters of a kth order HMM is proportional to the size of the corpus multiplying the $(k + 1)$th power of the hidden state count, while the time complexity of the deep learning used in GloVe is proportional to the square of the vector dimension of the corpus.

1.2 Problem Statement and Our Contributions

We first give the background of the problem. We want to create large, open domain knowledge-bases, or taxonomies, whose scale or coverage is especially important to the applications built on top of them. Because manually constructed taxonomies cannot reach sufficient scale and coverage, most of recent works (Hoffart et al. 2011; Ponzetto and Strube 2007; Wu et al. 2012) use data-driven approaches to automatically acquire taxonomies from large corpus such as the World Wide Web. Let us consider the open source Probase[7] (Wu et al. 2012) as an example. It contains millions of entities and classes, and the backbone of the taxonomy is the isA relationship. Besides isA relationship, there are many other relationships such as headquartered-in and is-CEO-of. Since the data

[7] http://research.microsoft.com/en-us/projects/probase/release.aspx.

and the relationships in the taxonomy are acquired from a huge Web corpus through syntactic-based IE, naturally there are many errors. Hence, data cleaning, especially on the isA relationships is extremely important for using the taxonomy.

In this article, our goal is to tackle the following two problems. The first problem is how to verify the correctness of hundred of millions of instances of isA relationships. That is, given a candidate pair $\langle c, e \rangle$, where c is a class and e is a candidate entity of the class, we want to evaluate how likely e is an entity of class c. The second one is how to verify the correctness of binary relationships, that is, given a candidate pair $\langle e_1, e_2 \rangle$, and a known relationship R between classes c_1 and c_2, we want to evaluate whether relationship R exists between e_1 and e_2. We now analyze the challenges in the tasks. The first challenge is the scale. For example, Probase contains 2.7 million categories and 5.5 million entities. It is impossible to learn a generative model (such as the HMM model) using contexts of all entities, because it is very time-consuming. The second challenge lies in improving the effectiveness of the verifier. As we mentioned, the feature representation based on contexts of words is very different from that based on contexts of entities. Meanwhile, neither a bag-of-words nor a set of hidden states can provide good semantics to understand the relationship between a candidate pair.

Motivated by this, we introduce a semantic, efficient, and effective approach for sparse extraction assessment in this article. And, our contributions are as follows.

First, we introduce a semantic approach to solve the aforementioned two problems. More precisely, we come up with a semantic representation of the contexts. This approach is natural because we are dealing with a large semantic network, which provides semantic information in various aspects. Using these information, we are able to introduce semantic features to describe a context, which leads to a lightweight and effective solution to context learning.

Second, we scan billions of Web documents using MapReduce[8] to capture the contexts of millions of entities and pairs of entities in Probase, and then compare the similarity between their contexts and the contexts of seeds.[9] We further use the similarity evaluated by our three semantic context-based approaches to represent the feature space given a pair, and then we train a classifier on a small number of labeled data varying with different base classifiers to select the best one for predicting sparse extractions. Extensive studies show that our approach can achieve better performance than state-of-the-art approaches in sparse extraction assessment.

1.3 Article Organization

The rest of the article is organized below. Section 2 introduces the related work in the assessment of IEs. Section 3 describes several syntactic and semantic approaches for context representation. Section 4 discusses how we acquire open domain knowledge and perform the conceptualization. Section 5 summarizes our approach. Section 6 presents experimental results. We give the conclusions in Section 7.

2 RELATED WORK

In this section, we give a brief introduction to some related work in the assessment of IEs. Researchers mainly make efforts from the following four dimensions to assess the quality of IE.

Context-based model building approach. In terms of the distributional hypothesis, this kind of approaches builds models using various contexts such as lexical and syntactic contexts, or semantic contexts to assess sparse extractions. Main works are summarized below. Zhou et al. proposed a tree kernel-based method (Zhou et al. 2010) with rich syntactic and semantic information for

[8] https://en.wikipedia.org/wiki/MapReduce.

[9] Seeds indicate some popular entities belonging to a given concept or a given relation.

the extraction of semantic relations between named entities. Yates et al. designed the TextRunner (Yates et al. 2007) system using the raw number of facts for Open IE (Etzioni et al. 2011). Dalvi proposed an open-domain IE method (Dalvi et al. 2012) for extracting class-entity pairs using Hearst patterns from HTML tables. Downey et al. proposed a language modeling-based method called REALM (Ahuja and Downey 2010; Downey et al. 2007) for assessing sparse extractions. It utilizes all sentences crawled from Web pages as the context to build HMM models and *n*-gram-based language models. Corro and Gemulla proposed a novel, clause-based approach called ClausIE (Corro and Gemulla 2013) to open IE, which extracts relations and their arguments from a natural language text based on dependency parsing and a small set of domain-independent lexica. Pennington et al. proposed a feature representation approach (Pennington et al. 2014) based on the word–word co-occurrence statistics and the word2vec tool. Oramas et al. used syntactic rules based on part of speech tags to extract entities and relations from unstructured music text sources as a knowledge graph (Oramas et al. 2015). Leng and Jiang proposed a deep learning approach (Leng and Jiang 2016) based on a stacked denoising auto-encoder on sentence-level features to extract manufacturing relationships underlying the text-based context. Existing cleaning operations are defined in an algorithmic way, and hence it is not clear how to extend the built-in operations without requiring low-level coding of internal or external functions. Fagin et al. proposed a rule-based approach (Fagin et al. 2016) to IE, which embarks on the establishment of a framework for declarative cleaning of inconsistencies in IE through principles of database theory. Cesare et al. presented a machine learning filter (Cesare et al. 2016) to enhance the precision of relation extractors while minimizing the impact on recall. It aims at filtering relation extractors' output using a binary classifier based on a wide array of features including syntactic, lexical and statistical features. Ghali and Qadi proposed a context-aware query expansion approach (Ghali and Qadi 2017) using language models and latent semantic analyses. Okamoto et al. applied a machine learning-based IE approach (Okamoto et al. 2017) to grasp the patent claim structure. It uses Markov logic network-based inference and distant supervision-based labeling to extract relations from patent texts.

In general, the above approaches using semantic contexts in the modeling outperform those using lexical and syntactic contexts in the tackling of sparse extractions. However, the feature space of texts in many approaches is represented using the distribution of a set of hidden states or a set of words instead of entities. Thus, it probably leads to a worse performance in the assessment of sparse extractions (See analyses in the fourth paragraph of Section 1.1 and in the last paragraph of Section 6.3). This is because each entity in the text can be an arbitrary multi-word expression instead of just a single word. Meanwhile, it is time-consuming for most of machine learning-based methods that require labeled data and model learning.

Heuristic-based approach. This kind of approaches usually requires a set of seed entities or some prior label distributional knowledge to identify IEs. Some representative works are as follows. Feldman and Rosenfeld designed a Web relation extraction system called URES (Feldman and Rosenfeld 2006). It learns powerful extraction patterns from unlabeled texts, using short descriptions of the target relations and their attributes. Zhu et al. developed a bootstrapping system called statistical Snowball (StatSnowball) (Zhu et al. 2009) based on the Snowball system (Agichtein and Gravano 2000). Zhang et al. proposed a graph-based framework (Zhang et al. 2015) to simultaneously learn the types of both entities and auxiliary signals. Alkan and Karagoz presented a new approach (Alkan and Karagoz 2016) based on a user-defined scoring mechanism so as to extract patterns from Web log data. Amal et al. described a Graph-Based Entity Profiling system called GBEP (Amal et al. 2017), which extracts information about persons of interest from the Web to construct a joint social graph, and uses it to obtain initial positive results for recommending related conference participants to each other. Ratner designed a new system for quickly creating, managing, and modeling training sets called Snorkel (Ratner et al. 2017). It enables users to

generate large volumes of training data by writing labeling functions, which are simple functions that express heuristics and other weak supervision strategies. Hanafi et al. presented a system that learns IE rules from a small set of user examples using Visual Annotation Query Language, called SEER (Hanafi et al. 2017). It is a less time-consuming alternative to the aforementioned machine learning methods, because these machine learning methods require large labeled datasets or rule-based approaches that are labor-intensive. Gong et al. proposed to learn multimodal rules (Gong et al. 2017) to improve the reliability of syntactic rules for text IE. The proposed system takes unannotated raw Web pages and a handful of seed instances as inputs, then automatically extracts information in a self-supervised manner, minimizing the human intervention. Reinanda et al. proposed a document filtering method (Reinanda et al. 2016) for long-tail entities. It is built on the hypothesis that there is a rich set of intrinsic features, based on aspects, relations, and the timeliness of the facts or events mentioned in the documents. Kejriwal and Szekely proposed a lightweight, feature-agnostic IE paradigm (Kejriwal and Szekely 2017) specifically designed for illicit domains. The proposed approach uses raw, unlabeled text from an initial corpus, and a few seed annotations per domain-specific attribute, to learn robust IE models for unobserved pages and websites.

However, all aforementioned approaches require previously preparing a set of seed entities given a relation or some prior label distributional knowledge, and require iterative operations to identify extraction patterns for the relation. Thus, it probably leads to erroneous entities due to dubious extraction patterns brought by random extraction errors in the iteration.

Redundancy-based approach. This kind of approaches uses a small set of patterns to find the redundant extractions. For instances, to improve the KNOWITALL's recall and extraction rate without sacrificing precision, Etzioni et al. presented a refined KNOWITALL system (Etzioni et al. 2005) based on the pointwise mutual information (Church and Hanks 1990). Downey et al. introduced a combinatorial "balls-andurns" model called URNS (Downeya et al. 2010). It computes the impact of sample size, redundancy, and corroboration from multiple distinct extraction rules on the probability that an extraction is correct. Gulhane et al. exploited content redundancy on the Web to extract structured data from template-based websites, and developed an efficient Apriori-style algorithm (Gulhane et al. 2010) to systematically enumerate attribute position configurations with sufficient matching values across pages. Unlike the heuristic-based approaches, these redundancy-based approaches utilize lexical and syntactic contexts or the distribution of the target and error sets to build models, but they are not good at assessing which extraction is more likely to be correct for sparse extractions.

Knowledge-based approach. This kind of approaches is built on some external resources to assess IEs. We can divide it into the following three categories according to the used knowledge bases. First, several representative works based on Wikipedia/Freebase knowledge bases are below. Gabrilovich and Markovitch proposed the explicit semantic analysis (ESA) approach (Gabrilovich and Markovitch 2009) using Wikipedia-based classes, for fine-grained semantic interpretation of unrestricted natural language texts. Wu and Weld proposed a Wikipedia-based open IE system called WOE (Wu and Weld 2010). Lee et al. developed a framework (Lee et al. 2011) that relies on the interconnections of the data in the taxonomies as well as in external data sources for entity resolution. Hoffart et al. developed a novel notion of semantic relatedness between two entities represented as sets of weighted keyphrases, called the Keyphrase Overlap RElatedness measure (KORE) (Hoffart et al. 2012). It focuses on disambiguating names in a Web or text document by jointly mapping all names onto semantically related entities registered in a knowledge base. To extract useful information from Wikipedia, Radhakrishnan and Varma proposed a method (Radhakrishnan and Varma 2013) using semantic features derived from Wikipedia categories. Meijer et al. presented a semantic approach (Meijer et al. 2014) based on Freebase for the automatic building of a domain taxonomy from text corpora. To tackle the issue of linkless knowledge base

in named entity disambiguation, Li et al. proposed a generative model (Li et al. 2016) to leverage useful disambiguation evidence scattered across the reference knowledge base such as Wikipedia. Yan and Jin presented a new text mining approach (Yan and Jin 2017) incorporating Wikipedia to the search scenario of detecting potential semantic relationships between topics. Lin et al. proposed a new knowledge graph model called KR-EAR (Lin et al. 2016) with entities, attributes, and relations using the Freebase knowledge base.

Second, representative works based on domain-knowledge are as follows. Sendi and Omri proposed a new approximate model (Sendi and Omri 2015) based on possibilistic networks, statistical computing and semantic proximity for extracting biomedical concepts using the MeSH (Medical Subject Headings) thesaurus.[10] To extract chemical-disease relations from PubMed[11] abstracts, Alam et al. proposed a general-purpose approach (Alam et al. 2016) based on machine learning techniques integrated with a limited number of domain-specific knowledge resources. Lima et al. presented OntoILPER (Lima et al. 2017) a system for extracting entity and relation instances from unstructured texts using ontology and inductive logic programming.

Third, representative works based on knowledge bases and Web data are as follows. Kondreddi et al. presented a novel system architecture, called Higgins (Kondreddi et al. 2014) to integrate an IE engine and a human computing engine using the resources like WordNet, ConceptNet, and Web data. Mausam et al. designed a novel Open IE extractor using bootstrapped dependency paths (called OLLIE) (Schmitz et al. 2012) based on Wikipedia and Web data. Taneva and Weikum proposed an approach (Taneva and Weikum 2013) that automatically extracts key contents from the Web for given input entities. It generates salient contents given an entity using minimal assumptions about the underlying sources. Jin et al. built a web-scale entity linking system (Jin et al. 2014) for tail entities on the Web. This system uses a posterior probability pursuit to exploit the sparse nature of entity linking, and uses the phrase unigram language model to effectively capture high-order dependencies among words. Gashteovski et al. proposed an Open IE called MinIE (Gashteovski et al. 2017). It follows OLLIE, but adds semantic annotations that make the extraction itself more compact and useful.

Most of aforementioned approaches are built on the semantic information in knowledge databases, such as Freebase and Wikipedia, but they still have a limited scale and coverage in terms of class space. Unlike the aforementioned approaches, our approach uses semantic context for assessing sparse extractions, and it is lightweight and supports Web scale data. To improve the accuracy, we aggregate three different semantic contexts to increase the data redundancy, and select an optimal classification model for assessing sparse extractions. All semantic contexts are extracted from the Web by specified patterns and preprocessed with the help of an open data source Probase. Authors in Wu et al. (2012) addressed that the scale of Probase is one order of magnitude larger than the previously known large corpus. Thus, our approach is more effective and efficient in assessing sparse extractions compared to the aforementioned knowledge-based approaches.

3 CONTEXT REPRESENTATION

In this section, we present several approaches for context representation, and we show that semantic approaches are superior in performance.

3.1 Modeling Contexts

The core tasks of the two problems, we are addressing in this article, are context representation and context comparison. Considering the first task, that is, given a pair $\langle c, e \rangle$, it aims to find $T(e)$,

[10] https://www.nlm.nih.gov/mesh/.
[11] https://www.ncbi.nlm.nih.gov/pubmed.

the context of e, and then compare it against the context of the seed entities of c to verify whether e is an entity of class c. This can be formulated as follows:

$$f(e, c) = \text{sim}(T(s_0), T(e)), \tag{1}$$

$s.t.,\ s_0 = argmax_{s \in S_c} sim(T(s), T(e))$, where S_c denotes seed entities of c, namely popular entities of c, and $sim(\cdot)$ is a similarity function. Clearly, the definition of $sim(\cdot)$ also relies on the representation of the context. We consider e as an entity of c if the value of $f(e, c)$ is beyond a certain threshold.

For the second task, that is, given a pair $\langle e_1, e_2 \rangle$ and a known relationship R, it aims to verify whether $\langle e_1, e_2 \rangle$ satisfies the relationship, the challenge is similar. Assume we know the relationship R is between classes c_1 and c_2. We do the following. First, we use Equation (1) to check whether e_1 and e_2 are entities of classes c_1 and c_2, respectively. If so, then, for any pair $\langle e_1, e_2 \rangle$, we find $T'(e_1, e_2)$, the context of e_1 and e_2 occurring together. Finally, we use Equation (2) to verify whether e_1 and e_2 satisfy R (S_R denotes the set of seed pairs of relationship R).

$$f'(e_1, e_2, R) = \text{sim}(T'(s_x, s_y), T'(e_1, e_2)), \tag{2}$$

$s.t.,\ < s_x, s_y >= argmax_{\langle s_1, s_2 \rangle \in S_R} sim(T'(s_1, s_2), T'(e_1, e_2))$.

It is clear that our primary task is to define $T(e)$ and $T'(e_1, e_2)$ in Equations (1) and (2). For a given sentence s, let us define w_s^e to be the text in a fixed-size window centered at e in sentence s, and $w_s^{e_1, e_2}$ to be the text in a fixed-size window centered at the middle of e_1 and e_2 in sentence s. Note that we exclude e from w_s^e and e_1, e_2 from $w_s^{e_1, e_2}$. Then, for a corpus that contains sentences $s_1, \ldots, s_n$, we can define context $T(e)$ and $T'(e_1, e_2)$ as

$$T(e) = \texttt{ContextExtract}(w_{s_1}^e, \ldots, w_{s_n}^e) \tag{3}$$

$$T'(e_1, e_2) = \texttt{ContextExtract}(w_{s_1}^{e_1, e_2}, \ldots, w_{s_n}^{e_1, e_2}), \tag{4}$$

where `ContextExtract` is a function that extracts context from text strings. Thus, we have unified the two problems into a single task, which is to define `ContextExtract`. In the following, we discuss several approaches for defining `ContextExtract`.

3.2 Syntactic Contexts

There are many different ways to use syntactic features to represent a context. In this subsection, we introduce bag-of-words and bag-of-neighboring-bigrams as two baseline methods for context representation.

Bag-of-words context. Given an entity e or a pair of entities $\langle e_1, e_2 \rangle$, sentences containing e or e_1 and e_2 are first collected, and words either from a fixed-sized window centered at the entities or from the entire sentences are obtained. And, then a vector $\{(t_i, w_i^t)\}$ is derived, where t_i indicates a word and w_i^t indicates its importance in terms of the tf-idf score, which is computed by the word statistics given the corpus. In the same way, context vectors for seeds and pairs of seeds are obtained, that is, we select top-k entities of the given class as the *seeds* according to their frequencies. For example, frequent entities in Table 1 are good seeds for class *country*. Finally, the cosine function is used to compare their similarity.

Bag-of-neighboring-bigrams context. Instead of using single words, we use bigrams that are immediate neighbors of e, e_1, or e_2. For example, suppose e occurs in the context of "located in e" or "Republic of e," we collect bigrams "located in" and "Republic of" as context for e. Alternatively, we can also use trigrams instead of bigrams. The context is represented as a vector $\{(p_i, w_i^p)\}$, where p_i is a bigram pattern and w_i^p is its weight. We do this for seeds and pairs of seeds as well, and we use the cosine function to compare the similarity of two vectorized concepts.

Table 2. Top 10 Classes of *Rhodesia* and *India*

Class of *Rhodesia*	Weight	Class of *India*	weight
British colony	0.102	*Country*	0.121
Country	0.085	Democracy	0.016
Great country	0.030	*Place*	0.015
Beautiful country	0.021	Signatory	0.014
Place	0.021	Land	0.014
Landlocked country	0.021	Market	0.0131
Colony	0.017	*Nation*	0.013
Nation	0.017	Big country	0.012
Threat	0.017	Large country	0.011
African country	0.013	Developing country	0.011

3.3 Semantic Contexts

Syntactic contexts are easy to obtain, but they are unstructured, and they are often noisy and confusing. Context model building-based approaches use a more structured representation of the context, but it is costly to obtain. In this article, we introduce a lightweight semantic representation for contexts.

isA-based context representation. Consider the example of *Rhodesia*. If we can get the classes *Rhodesia* belongs to, it is easy to judge whether *Rhodesia* is a country by comparing the class contexts of *Rhodesia* and the seeds in country. Table 2 shows the top 10 classes *India* and *Rhodesia* belong to respectively. We can see that the two lists have certain similarity. In other words, our isA-based context representation relies on the isA data, which is derived mostly from Hearst patterns. We will describe the resource of isA data later.

To implement our isA-based context representation, we require having the following data: (i) the seed entities of c for each class c; (ii) the set of classes that any entity e belongs to. For example, India may belong to classes such as country, developing country, democracy, and the like. (iii) For any pair of entity e and class c, we know how typical c is as a class for e. For entity, people may think of Arnold Schwarzenegger as a movie star, a politician, a bodybuilder, a businessman, or an investor. But the weight (typicality) of Arnold Schwarzenegger being a movie star is higher than being an investor.

From the above information, the vector for class c is derived below: $\mathcal{I}_c = \langle w_1^{\mathcal{I}}, \ldots, w_{k_{\mathcal{I}}}^{\mathcal{I}} \rangle$, where $w_i^{\mathcal{I}}$ is defined as $w_i^{\mathcal{I}} = \sum_{e \in S_c} p(c_i|e)$ $(1 \leq i \leq k_{\mathcal{I}})$, $k_{\mathcal{I}}$ indicates the size of top $k_{\mathcal{I}}$ classes e belongs to by the co-occurrences of e and c, S_c indicates the set of seed entities of c, and $p(c_i|e)$ indicates the typicality score of class c_i given entity e, that is, how typical c_i is among all the classes e belongs to. Next, for each extraction $\langle c, e \rangle$, the vector $\mathcal{I}_e = \langle w_1^{\mathcal{I}'}, \ldots, w_{k_{\mathcal{I}}}^{\mathcal{I}'} \rangle$ is derived, where $w_i^{\mathcal{I}'} = p(c_i|e)$ $(1 \leq i \leq k_{\mathcal{I}})$. Finally, a similarity function is used to decide how likely e is an entity of c

$$f_{isa}(c, e) = \text{sim}(\mathcal{I}_c, \mathcal{I}_e). \tag{5}$$

Attribute-based context representation. We use *Rhodesia* as an example again. There is no sufficient evidence to support or refute the claim that Rhodesia is a country using the Hearst patterns. However, when a country is talked by people, no matter big or small, rich or poor, some things likely will be mentioned, such as *capital city*, *president*, *congress*, *currency*, whether it is a *republic* or a *kingdom*, and the like. In other words, if the context of *Rhodesia* is represented by the presence of such entities, then it will be clear whether Rhodesia appears in contexts where real countries appear. As a matter of fact, the pattern "Republic of Rhodesia" appears 50 to 100 times more

Table 3. Top 10 Attributes of *Rhodesia* and *India*

Attributes of *Rhodesia*	Weight	Attributes of *India*	Weight
Republic	0.102	*Government*	0.105
New country	0.048	*People*	0.029
Prime minister	0.041	Reserve bank	0.028
African state	0.038	Constitution	0.026
Government	0.034	*Capital*	0.015
Star	0.024	President	0.013
People	0.020	Population	0.013
Liberation	0.015	Industrial development	0.013
Capital	0.014	Times	0.012
History	0.013	*Republic*	0.011

frequently in the Web corpus than the pattern "countries such as Rhodesia." Thus, our attribute-based context representation depends on the pattern of "the $\langle a \rangle$ of e is" (a indicates an attribute), which we will describe later.

Assume we are given a set of attributes $\{a_1, \ldots, a_n\}$ for any class c. For example, capital city, GDP, population are possible attributes for country. We are also given seed entities of class c. For example, the seed entities of country could be USA, Japan, Germany, China, and the like. We can obtain a vector for any class c in the form of $A_c = \langle w_1^A, \ldots, w_{k_A}^A \rangle$, where w_i^A $(1 \leq i \leq k_A)$ indicates how frequently seed entities of c in S_c appear together with attribute a_i of c in a corpus. It is defined as $w_i^A = \sum_{e \in S_c} p(a_i|e)$, where S_c is the set of seed entities of c, and $p(a_i|e)$ is the typicality score for e and attribute a_i.

To determine whether an entity e is of type c, the syntactic pattern "the $\langle a \rangle$ of e is" is used to obtain each candidate attribute $\langle a \rangle$ of e from the Web corpus. In this case, we can get a vector for e, namely $A_e = \langle w_1^{A'}, \ldots, w_{k_A}^{A'} \rangle$, where $w_i^{A'}$ $(1 \leq i \leq k_A)$ refers to the frequency a_i appears in the pattern with e. Finally, a similarity function is utilized to decide how likely e is an entity of c

$$f_{att}(c, e) = \text{sim}(A_c, A_e). \tag{6}$$

Table 3 lists the top 10 attributes of *India* as well as *Rhodesia* together with their normalized frequency. We can see that the two lists have a certain similarity.

Class-based context representation. The aforementioned isA-based and the attribute-based approaches still heavily rely on specific syntactic patterns, that is, the isA-based approach relies on the isA data, which is derived mostly from the Hearst patterns, while the attribute-based approach relies on the pattern of "the $\langle a \rangle$ of e is." The two approaches may bring in more evidence, but when we rely on fixed syntactic patterns, we limit ourselves to a much smaller corpus. This limits the amount of evidence we can find, especially when the extraction is about rare entities. In addition, such syntactic patterns may be appropriate for checking the type of a single entity (e.g., Rhodesia), but may not be appropriate if we want to decide whether a relationship exists between two arbitrary entities.

In our approach, instead of relying on fixed syntactic patterns, we map an arbitrary piece of text to a point in a semantic space, and then measure the distance in the semantic space between the point and points that correspond to the seed entities or seed pairs of entities. We call the technique conceptualization (Kim et al. 2013; Song et al. 2011), which simulates the process of human beings understanding things. Intuitively, given entities such as China and India, we "conceptualize" them

Table 4. Top 10 Conceptualized Classes

ID	Conceptualized class	Weight
1	Historical event	0.012
2	Country	0.011
3	Conflict	0.011
4	Partition maintenance operation	0.008
5	Liberal Jewish movement	0.008
6	Social structure	0.006
7	Security	0.006
8	Communist country	0.004
9	Merging market	0.004
10	Developing country	0.004

into a set of classes with the class of country ranked the highest; when given China, India, Russia, we "conceptualize" them into classes with emerging markets or BRIC[12] ranked the highest.

This enables us to process arbitrary piece of texts instead of texts that match fixed syntactic patterns. For example, consider the following sentence that contains the entity *Rhodesia*:

[1] He opposed the government's moves to restrict immigration, join the common market and reform the trade unions, was against the vietnam war and **Rhodesia**'s unilateral declaration of independence, and denounced the soviet suppression of "socialism with a human face" in czechoslovakia in 1968.

It is necessary to mention that the above sentence is selected from the given Web data by whether containing keywords "Rhodesia" and each seed entity of country such as "rhodesia india," or containing "rhodesia" and each seed attribute[13] of country, such as "rhodesia government" and "rhodesia federal republic." This text filtering aims to reduce the impact from texts with irrelevant classes. According to the above selected sentence, we underline the entities that can be found in Probase in the sentence. It is clear that the sentence does not match any of the syntactic patterns that are used to identify a country. Table 4 shows top 10 classes conceptualized on the sentence mentioned above (more details of conceptualization are given in Section 4.2). From this table, we can see that the country class is correctly conceptualized. This is because many entities in sentences are very relevant to the class of country. For example, Czechoslovakia, Vietnam, and Soviet are countries; government, common market, reform, socialism, trade unions are "properties" of countries, and the like. In the meanwhile, the other conceptualized classes also make sense, such as historical event, conflict, and security.

Formally, for an arbitrary piece of text, we conceptualize it to a set of conceptualized classes denoted as $\{cc_i\}_{i=1}^{k_C}$. To decide if an entity, say Rhodesia, is an entity of a class, say country, we perform conceptualization twice. First, we find the seed entities of the class, and collect the textual context of the seed entities in the Web corpus, and then we conceptualize the context to C_c, denoted as $C_c = \langle w_1^C, \ldots, w_{k_C}^C \rangle$, where w_i^C is the weight of the conceptualized class cc_i given the textual context of seed entities. Second, we find the textual context of the given entity denoted

[12]BRIC is a grouping acronym that refers to the countries of Brazil, Russia, India and China. https://en.wikipedia.org/wiki/BRIC.

[13]Seed attributes indicate top 10 attributes of a given class, which are selected from top 100 attributes of each seed entity of the given class according to the frequencies.

as e, and conceptualize it to C_e, denoted as $C_e = \langle w_1^{C'}, \ldots, w_{k_C}^{C'} \rangle$, where $w_i^{C'}$ is the weight of the conceptualized class cc_i given the textual context of e. Finally, we use a similarity function to decide whether the relationship holds

$$f_{con}(c, e) = \text{sim}(C_c, C_e). \tag{7}$$

3.4 Analysis

Before we describe how to enable the three semantic approaches we introduced above, we first get a feeling of how effective they are compared with the syntactic approach. We use the task of deciding whether an entity is a country as a test. The input is a set of 415 entities, 189 of which are of the type country. For each entity, each method gives a score in the range of [0,1], with score 1 meaning the entity is definitely a country, and score 0 meaning definitely not a country. We then group the entities by their ground truth, and sort the entities in each group by their scores. Figure 1 shows the result of four approaches mentioned above: bag-of-words (BM for short), attribute-based (AM for short), isA-based (IM for short), and class-based (CM for short).

Clearly, if the two curves are very close, then it means the method has little power in separating positive cases from negative ones. We define the *gain* of a method as the difference between the average scores given the positive and negative cases (denoted as P and N with $|P|$ and $|N|$ pairs, respectively), namely $Gain = \sum_{e_i \in P} score(e_i)/|P| - \sum_{e_i \in N} score(e_i)/|N|$. Clearly, the larger the gain, the more powerful the algorithm in separating the two cases. It is clear from Figure 1 that the bag-of-words approach is not effective. On the other hand, other approaches show big gaps between the positive curve and the negative curve.

4 KNOWLEDGE ACQUISITION AND CONCEPTUALIZATION

Three semantic context approaches have been described in the previous section, namely, attribute-based, isA-based, and class-based context representation. In order to support these approaches, it obviously requires the following two prerequisites, including common knowledge and the ability of conceptualization (Kim et al. 2013; Song et al. 2011). We hence focus on these two tasks in this section.

4.1 Knowledge Acquisition

The semantic approaches described in the previous section require the following open domain knowledge: (1) a large class space (including countries, pharmaceutical companies, etc.); (2) entities of each class (eg., China isA country); (3) attributes of each class (eg., population of country); (4) weights (typicality scores, e.g., $P(e|c)$).

Our approach is started with an open data source Probase,[14] which provides probabilistic isA knowledge for 2.7 million classes. The class space is big enough to cover almost every aspect of worldly facts. The isA relationships in Probase are harvested from 1.68 billion Web pages and 2 years' worth of Microsoft Bing's search log using syntactic patterns (e.g., the Hearst patterns (Hearst 1992)). For example, "… Asian countries such as China,…" serves as an evidence that China is of type Asian country. In this case, "China" is an entity (namely hyponym) while "Asian country" is a concept (namely hypernym). Furthermore, the isA knowledge in Probase comes with the weights that are needed in our work. For a class/entity pair $\langle c, e \rangle$, it provides two *typicality* scores: $P(e|c)$ and $P(c|e)$. The scores are known as typicality because, for

[14]This isA data is available at https://concept.research.microsoft.com/Home/Download. This data now contains up to 5,376,526 unique concepts, 12,501,527 unique instances, and 85,101,174 isA relations. In this article, we use a small version of Probase mentioned in Wu et al. (2012).

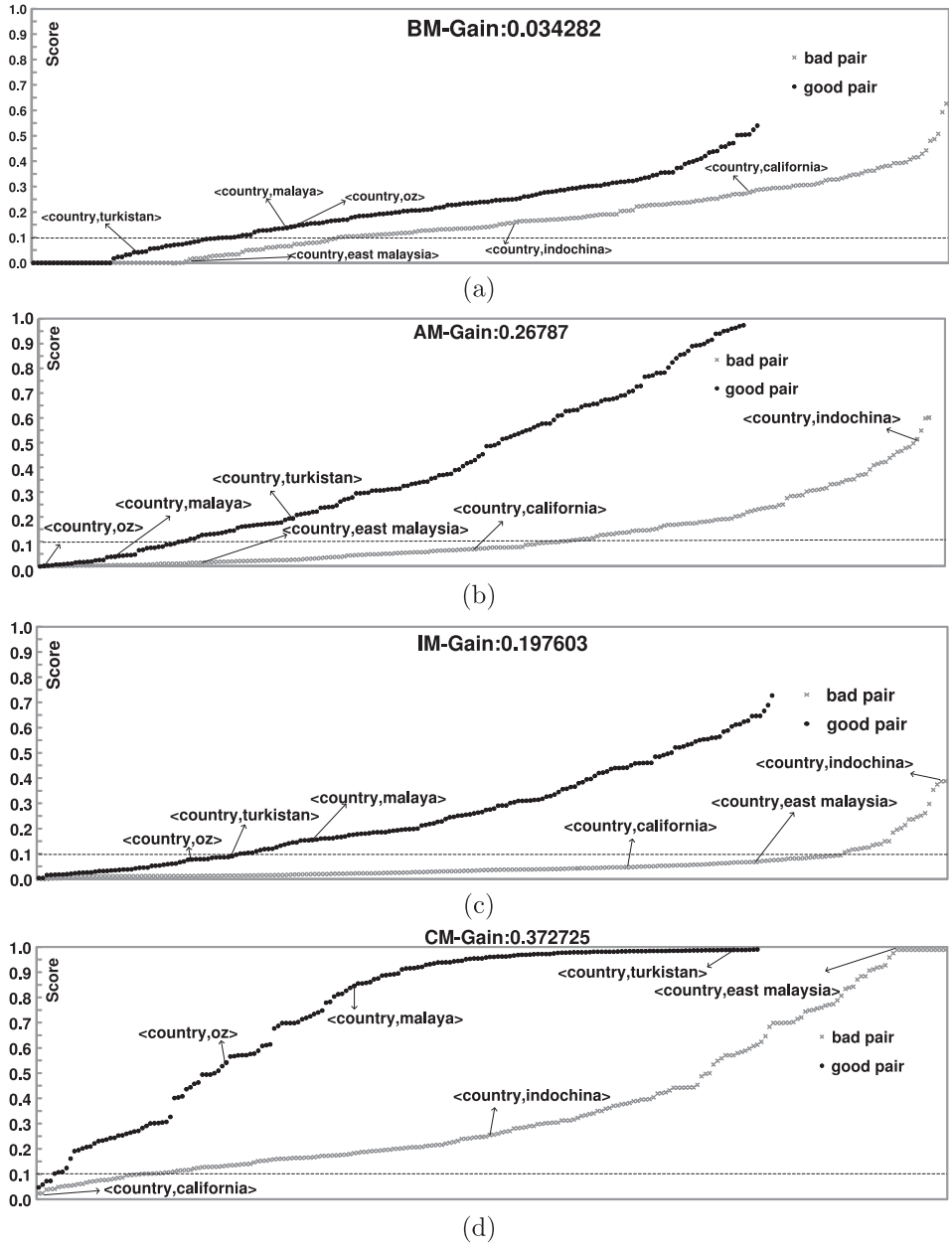

Fig. 1. Comparing four approaches in classifying entities as country or not country. (a) Bag-of-words-based context representation. (b) Attribute-based context representation. (c) isA-based context representation. (d) Class-based context representation.

example, $P(polarbear|mammal) > P(whale|mammal)$ because *polarbear* is more typical than *whale* as a *mammal*. Typicality score is derived as $P(e|c) = \mathrm{N}(c, e)/\mathrm{N}(c)$, where $\mathrm{N}(\cdot)$ indicates the occurrences of the given entities or entity pairs in Hearst extraction. In our approach, we also need to know the "seed" entities of each class. These are just entities of high typicality scores,

that is, entity e with $P(e|c)$ larger than a threshold. According to the open source Probase, we can easily get "a large concept space, seed instances of each class and seed instances of each entity" mentioned above.

In terms of the attribute extraction approach mentioned in Lee et al. (2013), we can obtain knowledge about attributes using Probase and the Web corpus. More specifically, for a given class c in Probase, and the seed entities $e_1, \ldots, e_k$ of c, we use the following syntactic pattern to derive attributes of c:

$$the\ \langle \boldsymbol{a} \rangle\ of\ (the/a/an)\ \langle \boldsymbol{e}_i \rangle\ is, \tag{8}$$

where $\boldsymbol{a}$ is an attribute and $\boldsymbol{e}_i$ is a seed entity of c. After obtaining all candidate attributes $\{\boldsymbol{a}\}$, we cluster and weight the attributes. This is necessary because an attribute may have many surface forms, for example, "date of birth," "birthdate," "birth date" and so on to represent the attribute "birthday." To do the clustering, we need positive evidence that indicates two surface forms are actually the same, and negative evidence that indicates two surface forms are definitely not the same. Regarding the positive evidence, we get it from sources such as Wikipedia Redirects and Wikipedia Internal Links. This is because accesses to Wikipedia URLs are redirected to other articles describing the same subject, and links to internal pages are expressed in shorthand by [Title|Surface Name] in Wikipedia, where Surface Name is the anchor text, and the page it links to is titled Title. Regarding the negative evidence, we get it from lists or tables. This is because two surface forms appear in the same list or as two columns in a table, and usually they do not mean the same thing. After collecting these evidence pairs, each connected component is taken as a synonymous attribute cluster. The most frequent attribute in each cluster is selected as a representative attribute of the synonymous attributes.

In addition, we evaluate the weights of attributes below. There are two important scores, including $P(a|e)$, the typicality of attribute a of entity e, and $P(e|a)$, the typicality of entity e for attribute a. The score is approximated by frequencies, namely $P(a|e) = \mathrm{N}(e, a)/\mathrm{N}(e)$, where $\mathrm{N}(\cdot)$ refers to the occurrences of given entities or entity pairs in pattern Equation (8).

4.2 Conceptualization

Conceptualization aims to derive classes hidden in the text. For example, from "India, China," we may derive the class of country, and from "India, Russia, China," the class of emerging market. In this article, we implement the conceptualization (Kim et al. 2013; Song et al. 2011) using the following three steps. First, given a piece of text (textual context of a single entity or a pair of entities), the knowledgebase is used as a dictionary to identify entities and concepts hidden in the text. Second, the most probably classes from the entities and attributes are derived instead of using the entites/attributes as a bag of words. That is, we find the distribution of classes C given entities/attributes $t_1, \ldots, t_k$, namely $P(C|t_1, \ldots, t_k)$. Third, a similarity function is adopted to compute the similarity between *clusters* of classes.

We know a single term t_i can be both an attribute and an entity. For instance, population is an attribute of the country class, but it can also be an entity of the geographical data class. In this case, an auxiliary variable z_i is used as an indicator for term t_i. More precisely, $z_i = 1$ if t_i is an entity, and $z_i = 0$ if t_i is an attribute. Meanwhile, a noisy-or model is used to infer the probability $P(c_k|t_i) = 1 - (1 - P(c_k|t_i, z_i = 1))(1 - P(c_k|t_i, z_i = 0))$, that is, term t_i invokes class c_k if it is an entity of c_k or it is an attribute of c_k. Here, we can get $P(c_k|t_i, z_i = 1) = P(c_k|e_i) = P(c_k, e_i)/P(e_i)$, where term t_i is regarded as an entity e_i, and $P(c_k|t_i, z_i = 0) = P(c_k|a_i) = \sum_{l:e_l \in E} P(c_k|e_l) \cdot P(e_l|a_i)$, where term t_i is regarded as an attribute a_i, and E is the set of entities that are related to attribute a_i and class c_k, and L indicates the size of E. Then, using the Naïve Bayes rule, we derive the class posterior

Table 5. Clusters of Top 10 Representative Classes

ID	Class of *Rhodesia*	ID	Class of *India*
1	*Country*	1	*Developed country*
1	*Nation*	1	*Nation*
1	Small country	1	*Country*
1	*Society*	1	Western nation
1	*Developed country*	1	Advanced country
1	Neighboring country	1	Overseas market
1	*Market*	1	*Market*
1	Muslim country	1	*Society*
2	Post-conflict country	1	State
2	War-torn country	1	English speaking country

given a set of terms denoted as T by Equation (9).

$$P(c_k|T) \propto P(c_k) \prod_i^L P(t_i|c_k) \propto \frac{\prod_i P(c_k|t_i)}{P(c_k)^{L-1}}. \tag{9}$$

Furthermore, Equation (7) is implemented to compare two sets of contexts after deriving the class distribution mentioned above, where context indicates a context in question and a typical context, and it is often derived from seed entities or seed entity pairs. Nevertheless, Probase has 2.7 million classes, in which many classes are correlated, e.g., "software company" and "IT company." Ignoring this correlation impairs the quality of the similarity function. We cluster the classes before comparing two class-based contexts. We use the refined k-Medoids clustering algorithm (Li et al. 2015) for this purpose. In sum, we use the above "content overlap" as the similarity function for clustering, that is, two classes are considered more similar if they have more identical entities.

One concept cluster can represent a sense or a general topic, recognized with its center concept. For example, for the cluster centered around company, most of its members are highly related to company, such as software company and technology company. Thus, "music star" and "pop star" will be highly similar, but "music star" and "railway worker" have smaller similarity. Each of the resulting cluster represents sort of a sense, and we apply Equation (7) over the senses, instead of the original classes. Table 5 lists the top 10 classes of *Rhodesia* and *India*, and their clustering results. The similarity of the two increased by 8% compared to the similarity evaluated on all conceptualized classes directly.

5 OUR APPROACH

In this section, we summarize our approach for the two tasks mentioned above, namely (i) the type checking: given a pair $\langle c, e\rangle$, evaluate how likely e is an entity of class c; and (ii) the relationship judgement: given a pair $\langle e_1, e_2\rangle$, and a known relationship R between classes c_1 and c_2, evaluate whether relationship R exists between e_1 and e_2.

Considering the type checking as shown in Algorithm 1, given a pair $\langle c, e\rangle$, we first use three semantic approaches of context representation, including attribute-based (Steps 1–5), isA-based (Steps 6–9), and Class-based (Steps 10–16) approaches to rank this extraction. Finally, we use the aforementioned attribute-based, isA-based, and Class-based scores to represent the feature space given the pair $\langle c, e\rangle$, and then we use Equation (10) to predict whether the given pair $\langle c, e\rangle$ is an isA relation as shown in Steps 17–19, called the AIC approach, where λ indicates the base classifier. If

ALGORITHM 1: Semantic Context-Based Type Checking

Input: $D = \{\langle c, e\rangle\}$: sparse information extractions;
Γ_{isA}: knowledgebase of isA relationships;
Γ_{sent}: the corpus of sentences;
Γ_{attr}: the attribute database;
Output: Similarity scores;

1 **for** *each sparse extraction* $\langle c,\ e\rangle \in D$ **do**
2 Select the seeds of the class c by the probabilistic score in Γ_{isA} as the seed set S_e;
3 Create the attribute vectors of c and e from Γ_{attr};
4 $f_{att}(c, e) \leftarrow$ Get the similarity score using Equation (6);
5 **end**
6 **for** *each sparse extraction* $\langle c,\ e\rangle \in D$ **do**
7 Create isA pattern vectors of c and e selectively from Γ_{isA};
8 $f_{isa}(c, e) \leftarrow$ Get the similarity score using Equation (5);
9 **end**
10 **for** *each sparse extraction* $\langle c,\ e\rangle \in D$ **do**
11 Collect the sentences containing the seed entity of c and the given entity e from Γ_{sent} respectively, denoted as ST_c and ST_e;
12 Identify all entities from ST_c and ST_e using Γ_{isA} and Γ_{attr};
13 Conceptualize all items for each sentence in ST_c and ST_e using Equation (9);
14 Create the class vectors of c and e denoted as C_c and C_e respectively;
15 $f_{con}(c, e) \leftarrow$ Get the similarity score using Equation (7);
16 **end**
17 **for** *each sparse extraction* $\langle c,\ e\rangle \in D$ **do**
18 $y^*_{\langle c,\ e\rangle} \leftarrow$ Get the classification result using Equation (10);
19 **end**

$y^*_{\langle c,\ e\rangle} = 1$ indicates the given pair $\langle c, e\rangle$ is a good pair, otherwise, it is a bad pair.

$$y^*_{\langle c,\ e\rangle} = \text{argmax}_{y\in\{0,1\}} P(y|\langle c,\ e\rangle, \lambda). \tag{10}$$

Considering the relationship validation as shown in Algorithm 2, it is based on the type checking in Algorithm 1. More specifically, given an entity–entity pair $\langle e_i,\ e_j\rangle$ and a known relationship R between classes c_i and c_j, we first repeat Algorithm 1 to check whether e_i and e_j are entities of classes c_i and c_j, respectively, in Step 2. If no, we can judge the given pair does not follow the R relationship. Otherwise, we collect the contexts containing the given pair $\langle e_i,\ e_j\rangle$ and each seed pair $\langle s_i,\ s_j\rangle$, respectively, and then we evaluate the probabilistic score that the pair $\langle e_i,\ e_j\rangle$ satisfies the known R relationship in Steps 3–11, where s_i and s_j is a seed in S_{c_i} and S_{c_j}, respectively. If the similarity is larger than the threshold α (eg. $\alpha = 0.3$), we consider the given relationship is true. Otherwise, it is false.

6 EXPERIMENTS

In this section, we first outline the experimental setup and then give the parameter analysis for several important parameters involved in our approach. Finally, we evaluate and analyze the semantic approaches of context representation as well as their effectiveness in the assessment of sparse extractions.

ALGORITHM 2: Relationship Validation

Input: $D = \{\langle e_i, e_j \rangle\}$: sparse information extractions;
R: the relationship between classes c_i and c_j;
Output: the similarity score of $\langle e_i, e_j \rangle$ satisfying R;

```
for each sparse extraction ⟨e_i, e_j⟩ ∈ D do
    Classify pairs ⟨c_i, e_i⟩ and ⟨c_j, e_j⟩ using Algorithm 1;
    if y*_⟨c_i, e_i⟩ == 1 and y*_⟨c_j, e_j⟩ == 1 then
        Collect the contexts of e_i and e_j occurring together as T'(e_i, e_j);
        Collect the seed sets of c_i and c_j as S_{c_i} = {s_i} and S_{c_j} = {s_j} respectively;
        for each seed pair ⟨s_i, s_j⟩ (s_i ∈ S_{c_i}, s_j ∈ S_{c_j}) do
            Collect the contexts of s_i and s_j occurring together as T'(s_i, s_j);
            Get the cosine score by comparing two vectorized contexts of T'(e_i, e_j) and T'(s_i, s_j);
        end
        Φ_R ← Get the similarity score satisfying the R relationship using Equation (2);
    end
end
```

6.1 Experiment Setting

The Web corpus we use contains 1.68 billion Web pages. Probase, which contains 2.7 million classes and 45 million pairs of relationships, is itself harvested from the corpus. We aim to clean Probase, that is, to identify false entities in a class and false pairs in a relationship. Given the scale of the problem, we perform data cleaning jobs on a map-reduce system with 10 machines. The cleaning process takes fewer than 10 hours.

Considering the experimental datasets, we randomly selected about 1,802 entities that have no more than 10 occurrences and belong to 12 single-word-based classes (such as country) in Probase as shown in Tables 6 and 8, and selected 2,277 entities that have no more than 5 occurrences and belong to 12 multi-word-based classes (such as middle east country) as shown in Tables 7 and 8. Each entity has no more than 10 occurrences in Hearst patterns and we call them sparse extractions. This is because more than 90% entities of the above 12 concepts have no more than 10 occurrences in Probase, namely lying in the long tail of the entity distribution curves. For example, Figure 2 shows the frequency distribution varying the number of entities in country. We can clearly see the long tail phenomenon under the dotted line with no more than 10 occurrences. We asked human judges to evaluate their correctness. We also looked into three binary relations: *isCapitalOf*, *isCurrencyOf*, and *headquarteredIn*. We randomly picked 315 sparse extractions between single-word-based classes (such as *headquarteredIn*(company, city) that have no more than 10 occurrences and 861 sparse extractions between multi-word-based classes (such as *headquarteredIn*(* company, * city)) that have no more than 5 occurrences, and we also picked the 10 most frequent extractions for each relation that serve as seeds. Details of all test relationships are shown in Tables 6 and 7. Meanwhile, we also give some examples of isA relationships and binary relationships as shown in Table 8.

Considering the competing approaches, we abbreviate all approaches involved in this article below, namely Bag-of-words-based (BM), Bag-of-Neighbouring-bigrams based (PM), Attribute-based (AM), Class-based (CM), isA-based (IM), and our final approach (called AIC). In addition, the other competing approaches contain the context-based model building approach called HMM-based (Ahuja and Downey 2010; Downey et al. 2007), and several knowledge-based approaches, including

Table 6. Datasets of isA and Binary Relationships with Only Single-Word-Based Classes Used in Experiments

Concepts	Total pairs in Probase	Pairs with freq. < 10	#Labeled pairs	#bad ones	#good ones
isA relationships with single-word-based classes					
Country	5,534	92.81%	415	226	189
Sport	2,866	92.18%	335	67	268
City	8,815	90.05%	231	33	198
Animal	5,562	92.38%	186	37	149
Seasoning	531	92.47%	169	41	128
Company	59,734	96.84%	82	9	73
Painter	1,097	98.09%	81	5	76
Currency	330	91.82%	78	8	70
Disease	8,280	92.60%	69	9	60
Film	10,859	96.62%	65	25	40
Language	2,703	93.53%	51	6	45
River	1,924	97.77%	40	2	38
Total	10,8235	95.25%	1,802	468	1,334
Binary relationships with single-word-based classes					
$isCapitalOf\langle country, city\rangle$			160	39	121
$isCurrencyOf\langle country, currency\rangle$			80	19	61
$headquarteredIn\langle company, city\rangle$			75	22	53
Total			315	80	235

the well-known IE system called OLLIE (Schmitz et al. 2012),[15] a new global log-bilinear regression model for word representation called GloVe (Pennington et al. 2014)[16] and the Knowledge-Graph-based models for representation learning called KR-EAR (Lin et al. 2016).[17] In approaches of GloVe and KR-EAR, a similarity score given a pair is obtained from the word vectors by first normalizing each feature across the vocabulary and then calculating the cosine similarity. We select the best performance of the GloVe approach using three corpora (with 6/42/840 billion tokens, respectively) obtained from Wikipedia dump and Web data, and the best performance of the KR-EAR (TransE) and KR-EAR (TransR) approaches as competing ones, called GloVe-best and KR-EAR-best, respectively.

For performance evaluation, we use the precision, recall and *F*-score on bad pairs and good ones as the evaluation measures, denoted as BP, BR, BF_1, GP, GR, and GF_1, respectively. For simplicity, the parameters of top $k_{\mathcal{I}}$ classes, top $k_{\mathcal{A}}$ attributes and top k_C conceptualized classes used in the semantic context representation are set to $k_{\mathcal{I}} = k_{\mathcal{A}} = k_C = 200$, respectively. Experiments are performed on an Intel Core 2 Duo CPU, 2.66-GHz PC with 4G main memory, running Windows 7 Enterprise. All experimental results reported in this section are averaged over five runs.

6.2 Parameter Estimation

In this subsection, we aim to select the optimum values of all important parameters involved in our approach, including the base classifier λ, the number of seeds $|S_c|$, and the similarity function

[15] http://openie.cs.washington.edu.
[16] http://nlp.stanford.edu/projects/glove/.
[17] https://github.com/thunlp/KR-EAR.

Table 7. Datasets of isA and Binary Relationships with Multi-Word-Based Classes Used in Experiments

Concepts	Total pairs in Probase	Pairs with freq. < 10	Concepts of labeled pairs	#Labeled pairs	#bad ones	#good ones
* country	28,434	96.36%	Asian country	74	38	36
			European country	90	44	46
			Middle east country	24	7	17
* sport	12,971	96.75%	Sea sport	26	5	21
			Winter sport	112	21	91
* city	21,063	98.16%	Action film	68	9	59
			American city	71	21	50
			Border city	51	15	36
* animal	17,980	97.43%	Aquatic animal	109	11	98
			Arctic animal	21	6	15
			Wild animal	434	41	393
* seasoning	553	100%	Italian seasoning	9	2	8
			Japanese seasoning	6	0	6
			Liquid seasoning	10	1	9
			Meat seasoning	4	1	3
			Mexican seasoning	6	0	6
* company	76,617	99.12%	Technical company	485	106	379
* painter	1,728	99.77%	Abstract painter	75	42	33
			American painter	41	3	38
			Australian painter	5	0	5
			Dutch painter	20	3	17
			German painter	9	3	6
* currency	958	98.75%	Asian currency	30	1	29
			European currency	21	2	19
			World currency	12	1	11
* disease	30,961	97.15%	Acute disease	96	21	75
			Bacterial disease	107	4	103
* film	11,956	99.69%	Action film	68	8	60
			American film	56	6	50
			Bollywood film	43	39	4
			Crime film	8	3	5
* language	9,927	97.71%	Body language	38	1	37
			Computer language	53	3	50
			Database language	6	1	5
			Development language	12	2	10
* river	1,810	99.83%	Freshwater river	9	0	9
			Perennial river	28	5	23
			Whitewater river	8	0	9
Total	214,958	98.07%	–	2,277	435	1,842
Binary relationships with multi-word-based classes						
$isCapitalOf\langle *country, *city\rangle$				188	100	88
$isCurrencyOf\langle *country, *currency\rangle$				188	100	88
$headquarteredIn\langle *company, *city\rangle$				485	180	305
Total				861	380	481

Table 8. Examples of isA Relationships and Binary Relationships

#bad pair	#good pair
On isA relationships with single-word-based classes	
<country, democratic people>	<country, g77>
<city, santa martha>	<city, amadora>
<sport, trafalgar park>	<sport, girls golf>
<animal, cauquenes>	<animal, moon snail>
<seasoning, bacon bit>	<seasoning, five spice>
<company, institute>	<company, hasbro>
<painter, robert young>	<painter, childe hassam>
<currency, australian>	<currency, american dollar>
<disease, addisons disease>	<disease,cystoid macular edema>
<film, forest gump>	<film, breach>
<language, francophone>	<language, micmac>
<river, manda>	<river, missouri river>
On isA relationships with multi-word-based classes	
<asian country,chinese taipei>	<asian country, islamic republic of pakistan>
<border city, man areas in florida>	<border city, san diego near california>
<winter sport, international federation of ice hockey>	<winter sport, dog sledding during the cold season>
<aquatic animal, seals being viewed from the shoreline>	<aquatic animal, insect larva>
<liquid seasoning, vanilla>	<liquid seasoning, gravy browning>
<technical company, seattle orthopedic in poulsbo>	<technical company, integrated data communications on bainbridge island>
<abstract painter, jan van eyck>	<abstract painter, abstract expressionist jackson pollock>
<european currency, swiess franc>	<european currency, dutch guilder>
<bacterial disease, chalcis obscurata>	<bacterial disease, pneumococcal disease from outbreak>
<action film, frame-by-frame drama>	<action film, showdown in little tokyo>
<body language, use of formal>	<body language, unusual tones of voice>
<perennial river, kwando-linyanti-chobe system>	<perennial river, ganges in india>
On binary relationships between single-word-based classes	
<dili, east timor>	<andorra, andorra la Vella>
<baht, thailand>	<colombia, colombian peso>
<espoo, general electric>	<michelin, clermont-ferrand>
On binary relationships between multi-word-based classes	
<data center operator, kenosha>	<sierra wireless, richmond>
<european country, ottawa>	<islamic republic of pakistan, islamabad>
<papua new guinia, papua new guinean kina>	<neighbor sweden, swedish krona>

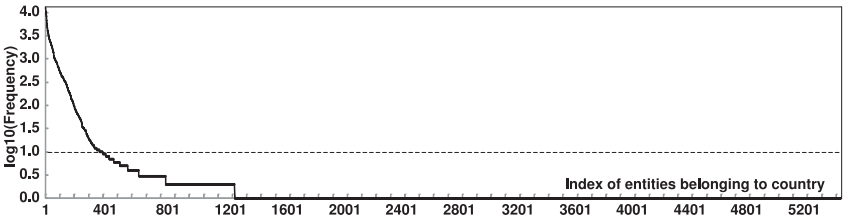

Fig. 2. Frequency distribution of entities in the class of country.

$sim(\cdot)$. All experiments are conducted in the cases varying with values of the specified parameter while keeping others unchanged. Details of parameter settings are as follows.

To conduct the performance analysis among all comparing base classifiers including J48, RandomForest, SMo based on PolyKennel and based on RBFKennel, Naïve Bayes, and Perceptron[18] systematically, we employ Friedman test (Demšar 2006) widely-accepted as the favorable statistical test for comparisons of multiple algorithms over a number of datasets (Zhang and Wu 2015). That is, given K comparing algorithms and N datasets, let $R_j = \frac{1}{N}\sum_{i=1}^{N} r_i^j$ denote the average rank for the jth algorithm, where r_i^j indicates the rank of the jth algorithm on the ith dataset. Under the null hypothesis, the Friedman statistic F_r is defined as $F_r = (N-1)\chi_F^2/(N(K-1)-\chi_F^2)$, $s.t.,\ \chi_F^2 = 12N/(K(K+1))[\sum_{j=1}^{K} R_j^2 - K(K+1)^2/4]$. We can get the value of F_r in the evaluation metric of WF_1[19] using different base classifiers, namely, $F_r = -5.315$ on 12 isA relationships with single-word-based classes, and $F_r = -6.743$ on 12 isA relationships with multi-word-based classes. At significance level $\alpha = 0.05$, the null hypothesis of "equal" performance among all base classifiers cannot be rejected because the value of F_r is lower than the corresponding critical value $F_{\frac{\alpha}{2}}(K-1,(K-1)(N-1))$=2.63 ($K = 7,\ N = 12$). Therefore, we get the conclusion that there is no significant difference among performances using different base classifiers. In this case, we select the logistic regression classifier with the best performance on WF_1 in Table 9 as the final classification model λ. Because from experimental results using 10 cross-validation on 12 isA relationships, we can see that logistic regression classifier can beat other competing classifiers on five isA relationships, and the average ranking is the first.

Regarding the number of seeds $|S_c|$, it is relevant to all syntactic and semantic approaches for context representation in this article. We take the prediction accuracy of our three semantic approaches (AM, CM, and IM) varying with values of $|S_c|$ from 1 to 50 for an example to show the selection on the optimal value of $|S_c|$. In the observation of experimental results as shown in Figure 3, we can see that as the values of $|S_c|$ increase, the prediction accuracy of theses three approaches is first increasing and then maintaining stably. In the meanwhile, we can get a high prediction accuracy if specifying $|S_c| \geq 10$. Therefore, in the following experiments, we select $|S_c| = 10$ as a candidate optimal value for all syntactic and semantic approaches for context representation.

Regarding the similarity function $sim(\cdot)$, we take the performance of AM varying with five similarity functions for an example to select the optimal function. All similarity functions include *cosine,*[20] *Jaccard,*[21] *JaccardExt*, JS_{sim}, and KL_{sim}, where *JaccardExt* indicates the Tanimoto

[18]These six base classifiers are from the open data mining software called Weka. http://www.cs.waikato.ac.nz/ml/weka/.
[19]In this article, WF_1 indicates the weighted average value of BF_1 and GF_1, that is, $WF_1 = BF_1 \cdot R(bad\ pairs) + GF_1 \cdot R(good\ pairs)$, where $R(bad\ pairs)$ and $R(good\ pairs)$ indicates the ratio of bad pairs and good pairs in given dataset. The larger the value of WF_1, the better the performance of the base classifier.
[20]https://en.wikipedia.org/wiki/Trigonometric_functions#cosine.
[21]https://en.wikipedia.org/wiki/Jaccard_index.

Table 9. Performance Comparison Using Different Base Classifiers

	J48	RandomForest	Logistic regression	SMO		Naive-Bayes	Perceptron
				PolyKennel	RBFKennel		
On 12 isA relationships with single-word-based classes							
animal	0.781(2)	0.719(4)	**0.789(1)**	0.614(6)	0.614(6)	0.743(3)	0.642(5)
city	0.872(4)	0.895(2)	**0.898(1)**	0.808(6)	0.808(6)	0.875(3)	0.832(5)
company	0.882(7)	**0.94(1)**	0.89(3)	0.89(3)	0.89(3)	0.933(2)	0.89(3)
country	0.778(6)	0.74(7)	**0.796(1)**	0.785(4)	0.784(5)	0.788(3)	0.794(2)
currency	0.829(2)	0.82(6)	**0.868(1)**	0.822(3)	0.822(3)	0.793(7)	0.822(3)
film	**0.914(1)**	0.774(5)	0.817(4)	0.821(3)	0.516(7)	0.84(2)	0.691(6)
disease	0.914(2)	0.901(4)	0.914(2)	0.841(5)	0.841(5)	**0.923(1)**	0.841(5)
language	0.763(7)	0.814(2)	**0.846(1)**	0.776(3)	0.776(3)	0.774(6)	0.776(3)
river	0.94(1)	0.877(7)	0.899(6)	**0.94(1)**	**0.94(1)**	0.94(1)	0.94(1)
seasoning	**0.851(1)**	0.832(2)	0.783(3)	0.649(5)	0.649(5)	0.783(3)	0.649(5)
sport	**0.863(1)**	0.806(2)	0.804(3)	0.767(5)	0.767(5)	0.797(4)	0.767(6)
painter	0.968(2)	0.968(2)	0.957(7)	0.968(2)	0.968(2)	**1(1)**	0.968(2)
On 12 isA relationships with multi-word-based classes							
* animal	0.861(3)	0.872(1)	0.844(5)	0.844(5)	0.844(5)	0.866(2)	0.852(4)
* city	**0.891(1)**	0.763(2)	0.731(3)	0.622(6)	0.622(6)	0.698(4)	0.668(5)
* company	0.820(4)	**0.834(1)**	0.829(3)	0.760(6)	0.760(6)	**0.834(1)**	0.787(5)
* country	**0.761(1)**	0.600(6)	0.730(2)	0.720(3)	0.354(7)	0.714(4)	0.712(5)
* currency	**0.816(1)**	**0.816(1)**	**0.816(1)**	**0.816(1)**	**0.816(1)**	**0.816(1)**	**0.816(1)**
* film	**0.883(1)**	0.866(7)	**0.883(1)**	**0.883(1)**	**0.883(1)**	0.873(6)	**0.883(1)**
* disease	0.900(3)	0.902(2)	0.893(4)	0.833(6)	0.833(6)	**0.908(1)**	0.870(5)
* language	**0.870(1)**	0.853(7)	0.861(6)	**0.870(1)**	**0.870(1)**	**0.870(1)**	**0.870(1)**
* river	**1.000(1)**	**1.000(1)**	**1.000(1)**	**1.000(1)**	**1.000(1)**	**1.000(1)**	**1.000(1)**
* seasoning	0.756(4)	0.756(4)	0.756(4)	**0.776(1)**	**0.776(1)**	0.645(7)	**0.776(1)**
* sport	0.602(3)	0.457(6)	**0.624(1)**	0.612(2)	0.256(7)	0.602(3)	0.529(5)
* painter	0.502(7)	0.589(2)	**0.608(1)**	0.556(3)	0.556(3)	0.552(5)	0.543(6)
Average ranking	2.750	3.500	**2.708**	3.792	4.000	3.000	3.542

Bold values indicate the best experimental results on each evaluation measure.

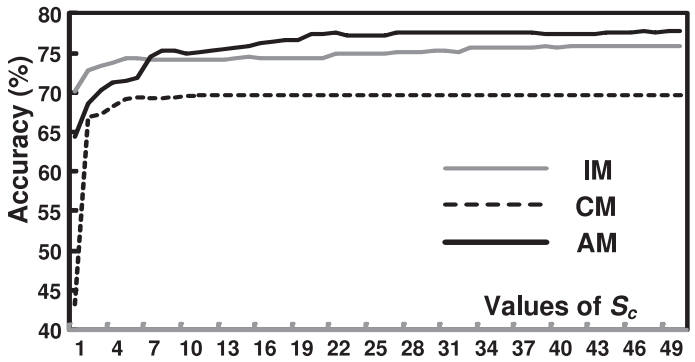

Fig. 3. Parameter setting on the value of $|S_c|$.

similarity,[22] JS_{sim} indicates the Jensen–Shannon divergence[23] based similarity, and KL_{sim} indicates the smoothed Kullback–Leibler divergence[24] based similarity. Figure 4 reports the experimental results of AM varying with the above five similarity functions. To simplify the compari-

[22] https://en.wikipedia.org/wiki/Jaccard_index#Tanimoto_coefficient.
[23] https://en.wikipedia.org/wiki/Jensen-Shannon_divergence.
[24] https://en.wikipedia.org/wiki/Kullback-Leibler_divergence.

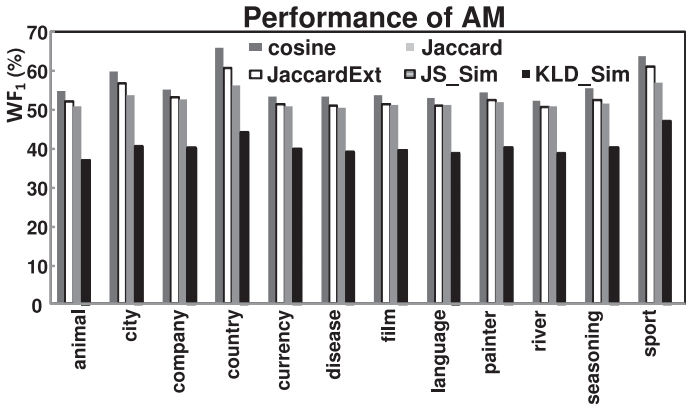

Fig. 4. Performance of attribute-based method varying with five similarity functions.

son, we also use the overall evaluation measure WF_1 here, that is, the larger the value of WF_1, the better the performance of the similarity function. From the experimental results, we can see that AM with the cosine similarity function beats all others, because the value of WF_1 is the highest. This conclusion is the same as that in the approaches of BM, PM, IM, and CM. Thus, in the following experiments, we select the cosine similarity function as $sim(\cdot)$ used in our methods.

6.3 Type Checking

We compare the performance of our AIC approach and eight competing ones in type checking, namely checking whether an entity is a true entity of a class. According to Friedman test mentioned above, we can get the values of F_r using our AIC approach and several competing approaches as shown in Table 10. And, we can see that at significance level $\alpha = 0.05$, the null hypothesis of "equal" performance among the comparing algorithms is clearly rejected in terms of each evaluation metric. Consequently, we employ the *Bonferroni–Dunn test* (Demšar 1961) to further analyze the relative performance among comparing approaches. We treat our AIC approach as the dominating approach, the difference between the average ranks of AIC and one comparing approach is compared with the following critical difference (CD): $CD = q_a\sqrt{K(K+1)/(6N)}$. For Bonferroni–Dunn test, we have CD =3.046 ($K = 9$, $N = 12$) on isA relationships and CD =5.38 ($K = 8$, $N = 3$) on binary relationships at significance level α =0.05. Accordingly, the performance between our approach and one comparing approach is deemed to be significantly different if their average ranks over all datasets differ by at least one CD. Figure 5 shows our AIC approach against all competing approaches with the *Bonferroni–Dunn test*. In this figure, we use the following method to get the ranking of each approach. For example, if our AIC approach performs best compared to other baseline ones on isA relationship of country regarding the evaluation measure GF_1, it is ranked in 1. Correspondingly, we can get all rankings of AIC on 12 isA relationships regarding the evaluation measure GF_1. Finally, we averaged these 12 rankings for our AIC approach as the final ranking. In a similar way, we can get the ranking of each approach, respectively, according to the experimental results on each evaluation measure. Thus, each approach has a ranking value from 1 (top 1) to 9 (top 9). At the meanwhile, we use a line to connect the approach without any significant difference according to the value of critical difference (CD). For example, in Figure 5(a), our AIC approach has the average ranking value of 2.3, while our AM approach has the average ranking value of 3.8 regarding the evaluation measure of GF_1. We compare the CD value with the difference between rankings of AM and AIC, and we find that it is lower than the value of CD (namely

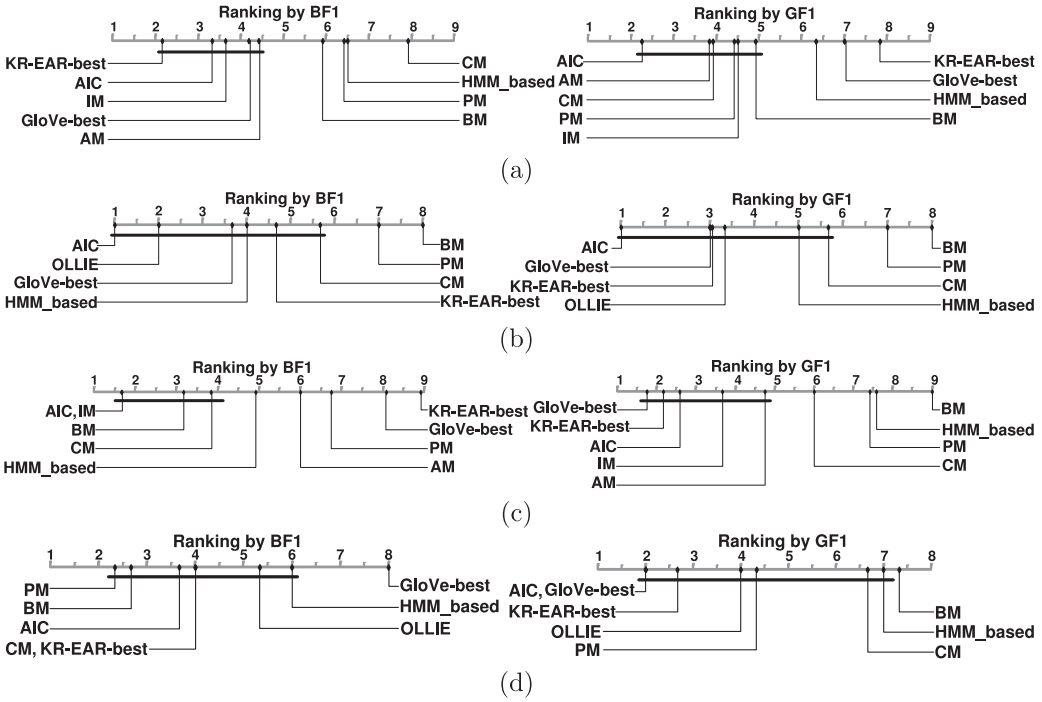

Fig. 5. Our AIC approach against all competing approaches with the *Bonferroni–Dunn test*. Approaches not connected with our AIC approach in the CD diagram are considered to have significantly different performance from the dominating approach (significant level $\alpha = 0.05$). (a) On 12 isA relationships with single-word-based classes. (b) On three binary relationships with single-word-based classes. (c) On 12 isA relationships with multi-word based classes. (d) On three binary relationships with multi-word-based classes.

Table 10. Friedman Statistics F_r and the Critical Value

Performance of our approach and competing algorithms			
On isA relationships with single-word based classes (F_r(left)) and with multi-word-based classes (F_r(right)), $K = 9, N = 12$			
Evaluation measure	F_r(left)	F_r(right)	Critical value ($\alpha = 0.05$)
BF_1	5.656	148.46	3.046
GF_1	7.592	119.73	
On three binary relationships with single-word based classes (F_r(left)) and with multi-word-based classes (F_r(right)), $K = 8, N = 3$			
Evaluation measure	F_r(left)	F_r(right)	Critical value ($\alpha = 0.05$)
BF_1	29.500	5.554	5.380
GF_1	20.235	10.194	

3.8−2.4 = 1.4 <3.046). In this case, we consider these two approaches have no significant difference, therefore, we use a line to connect.

Figure 5(a) and (c) illustrates the CD diagrams on isA relationships regarding the evaluation measures of BF_1 and GF_1, where the average rank of each competing approach is marked among the axis, namely lower ranks to the right. In each subfigure, any competing approach whose

average rank is within one CD to that of the best approach is interconnected with a thick line. From these figures, we can see that on isA relationships, our AIC approach is the dominating approach or competitive to the dominating approach regarding only BF_1 or GF_1. However, considering both evaluation measures of BF_1 and GF_1, our approaches of AIC and IM are comparable to each other, which significantly perform better than other six competing approaches. More details of experimental results are as follows.

Tables 11–14 show the prediction results on isA relationships (with single-word based and multi-word-based classes) between our AIC approach and five baseline ones. We can see the following from tables. First, as compared to the syntactic methods BM and PM, semantic methods win in 10 classes and lose in only 2 classes (company and river) considering the evaluation measures of BF_1 and GF_1. This is because the semantic information is conducive to improve the prediction performance compared to the syntactic information. Second, our AIC approach can correctly label more good pairs while labeling fewer bad pairs correctly than other three semantic methods. This is because our AIC approach is aggregated from the other three semantic context-based approaches.

Tables 15 and 16 report the performance of our AIC approach compared to other three competing approaches of HMM-based, Glove-best, and KR-EAR-best on isA relationships. We can observe the following. First, on all evaluation measures, our approach can beat HMM-based method in almost all 12 classes. This is because HMM-based method represents the feature space of texts using the distribution of a set of words and a set of hidden status, respectively. However, in most cases, each entity in the text can be an arbitrary multi-word expression instead of just a single word. Thus, a hidden status induced from the words can lead to a worse performance in the assessment of sparse extractions due to the missing semantic information. Second, our approach is superior or comparable to KR-EAR-best and Glove-best approaches regarding both evaluation measures of BF_1 and GF_1. The reason is analyzed below. Our approach and these two competing approaches are built on the knowledge bases of Probase and Freebase, respectively. It is beneficial to improve the performance in the assessment of sparse IE due to the introduction of semantic information. However, the difference lies in that Probase is more scalable and has a large coverage in terms of class space compared to Freebase. Thus, our approach is more competitive.

6.4 Relationship Validation

We now compare the effectiveness of semantic and syntactic approaches in the handling of sparse extractions with binary relationships. First, according to the CD diagrams in Figure 5(b) and (d), we can see that regarding the binary relationships, our AIC approach is also the dominating approach on both evaluation measures or is comparable to the dominating approach.

Furthermore, Figures 6 and 7 and Table 17 report the prediction performance of our AIC approach in Algorithm 2 compared to the competing approaches on three binary relationships between single-word based classes and between multi-word based classes. On one hand, according to ROC curves [25] from Figure 6 and experimental results on evaluation measures of BF_1 and GF_1 in Table 17, we observe that our approach can achieve better performance compared to the syntactic context-based approaches on three binary relationships no matter with single-word based classes or with multi-word based classes. These observations reveal that our approach is more effective by introducing the semantic context extracted from the Probase knowledge base. On the other hand, experimental results from Figure 7 and Table 17 show that our approach outperforms competing knowledge-based approaches regarding both evaluation measures of BF_1 and GF_1. This is because the Glove-best approach uses the word–word co-occurrence statistics to represent the

[25] We know that "the larger the area under the ROC curve, the better the performance for an algorithm" (Swets 1996).

Table 11. Comparison Between Our Approach and Five Baselines on 12 isA Relationships with Single-Word-Based Classes: Part1

isA	Method	BM	PM	CM	AM	IM	AIC
country	*BR*	11.0	34.6	7.1	39.0	**67.1**	57.0
	BP	71.4	68.2	78.9	78.1	84.5	**84.8**
	GR	94.5	80.9	**97.8**	86.8	85.2	88.9
	GP	46.0	50.8	47.2	54.1	**68.2**	65.5
	BF_1	19.0(5)	47.0(4)	13.1(6)	52.0(3)	**74.8**(1)	68.2(2)
	GF_1	61.9(6)	62.4(5)	**87.3**(1)	66.7(4)	75.8(2)	75.4(3)
sport	*BR*	19.4	20.0	7.4	73.1	**80.6**	62.8
	BP	41.9	26.5	19.0	**49.0**	44.3	42.9
	GR	93.3	84.8	**93.5**	81.0	74.6	84.4
	GP	82.2	79.4	83.0	92.3	**93.9**	92.4
	BF_1	26.5(4)	22.8(5)	10.7(6)	**58.7**(1)	57.1(2)	50.9(3)
	GF_1	87.4(2)	82.0(5)	31.6(6)	86.3(3)	83.2(4)	**88.2**(1)
city	*BR*	30.3	25.8	23.3	33.3	**60.6**	52.4
	BP	27.0	16.7	36.8	**73.3**	44.4	64.7
	GR	85.9	78.7	93.4	**98.0**	87.4	96.2
	GP	87.8	86.6	88.1	89.8	93.0	**93.9**
	BF_1	28.6(4)	20.3(6)	28.6(4)	45.8(3)	51.3(2)	**57.9**(1)
	GF_1	86.8(4)	82.5(5)	52.8(6)	93.7(2)	90.1(3)	**95.0**(1)
animal	*BR*	7.9	20.0	13.9	**39.5**	7.9	31.8
	BP	8.6	41.2	11.1	28.8	37.5	**70.0**
	GR	78.2	92.8	69.2	75.2	**96.6**	94.9
	GP	76.7	82.1	74.4	**83.0**	80.4	78.9
	BF_1	8.2(6)	26.9(3)	13.9(4)	33.3(2)	13.0(5)	**43.8**(1)
	GF_1	77.4(5)	87.1(2)	11.1(6)	78.9(4)	**87.8**(1)	86.2(3)
seasoning	*BR*	2.4	2.6	0.0	22.0	**92.7**	65.4
	BP	25.0	7.7	0.0	50.0	31.4	53.1
	GR	97.6	90.6	**98.4**	93.0	35.2	81.3
	GP	75.2	75.2	75.9	78.8	**93.8**	87.8
	BF_1	4.4(4)	3.8(5)	0.0(6)	30.5(3)	46.9(2)	**58.6**(1)
	GF_1	84.9(3)	82.1(5)	**85.7**(1)	85.3(2)	51.1(6)	84.4(4)
company	*BR*	**53.9**	18.2	0.0	20.0	20.0	20.0
	BP	25.9	20.0	0.0	16.7	25.0	**50.0**
	GR	76.5	90.0	95.2	86.3	91.8	**98.4**
	GP	91.6	88.9	92.3	88.7	89.3	**93.8**
	BF_1	**35.0**(1)	21.0(4)	0.0(6)	18.2(5)	22.2(3)	28.6(2)
	GF_1	83.3(6)	89.4(4)	93.8(2)	87.5(5)	90.5(3)	**96.1**(1)

Bold values indicate the best experimental results on each evaluation measure.

feature space; however, there are also some co-occurrences for bad pairs in the given corpus. Thus, it is insensitive to the bad sparse extractions, which causes a worse performance in the assessment of bad sparse extractions. Considering the other two competing approaches, OLLIE uses the information of Web data containing more noise, while KR-EAR-best uses the Freebase knowledge base, whose coverage is lower than ours. Meanwhile, our approach aggregates three semantic contexts using an optimal classification model, and it considers more evidence to assess sparse extractions.

Table 12. Comparison Between Our Approach and Five Baselines on 12 isA Relationships with Single-Word-Based Classes: Part 2

isA	Method	BM	PM	CM	AM	IM	AIC
painter	*BR*	20.0	60.0	25.0	0.0	**100.0**	0.0
	BP	9.1	15.0	7.7	0.0	**15.6**	0.0
	GR	86.8	77.6	83.8	**93.3**	64.5	91.3
	GP	94.3	96.7	95.4	94.6	**100.0**	**100.0**
	BF_1	12.5(3)	24.0(2)	11.8(4)	0.0(5)	**27.0**(1)	0.0(5)
	GF_1	90.4(3)	86.1(5)	89.2(4)	94.0(2)	78.4(6)	**95.5**(1)
currency	*BR*	12.5	12.5	14.3	62.5	**100.0**	83.3
	BP	25.0	5.3	**50.0**	31.3	21.1	26.3
	GR	95.5	73.5	**98.4**	84.3	57.1	77.4
	GP	90.0	87.7	91.3	95.2	**100.0**	98.0
	BF_1	16.7(5)	7.4(6)	22.2(4)	**41.7**(1)	34.8(3)	40.0(2)
	GF_1	92.7(2)	80.0(5)	**94.7**(1)	89.4(3)	72.7(6)	86.5(4)
disease	*BR*	37.5	12.5	16.7	**55.6**	44.4	50.0
	BP	21.4	16.7	12.5	62.5	28.6	**66.7**
	GR	80.0	91.4	86.3	95.0	83.3	**97.0**
	GP	89.8	88.3	89.8	93.4	90.9	**94.1**
	BF_1	27.3(4)	14.3(5)	14.3(5)	**58.8**(1)	34.8(3)	57.1(2)
	GF_1	84.6(6)	89.8(3)	88.0(4)	94.2(2)	87.0(5)	**95.5**(1)
film	*BR*	46.7	**96.6**	0.0	32.0	56.0	75.0
	BP	51.9	38.9	0.0	66.7	**77.8**	75.0
	GR	72.3	4.4	**100.0**	90.0	90.0	87.5
	GP	68.0	66.7	68.0	67.9	76.6	**87.5**
	BF_1	49.1(3)	22.3(5)	0.0(6)	43.2(4)	65.1(2)	**75.0**(1)
	GF_1	70.1(5)	8.2(6)	81.0(3)	77.4(4)	82.8(2)	**87.5**(1)
language	*BR*	39.0	50.0	0.0	50.0	**66.7**	**66.7**
	BP	78.1	42.9	0.0	16.7	28.6	**57.1**
	GR	86.8	90.2	**92.7**	66.7	77.8	90.6
	GP	54.1	92.5	86.4	90.9	**94.6**	93.5
	BF_1	52.0(2)	46.2(3)	0.0(6)	25.0(5)	40.0(4)	**61.5**(1)
	GF_1	66.7(6)	91.4(2)	89.4(3)	76.9(5)	85.4(4)	**92.1**(1)
river	*BR*	0.0	**50.0**	0.0	48.3	48.6	0.0
	BP	0.0	25.0	0.0	10.1	22.1	0.0
	GR	83.3	**92.1**	82.4	67.5	72.1	70.8
	GP	85.4	**97.2**	93.3	80.6	81.2	94.4
	BF_1	0.0(4)	**33.3**(1)	0.0(4)	16.8(3)	30.4(2)	0.0(4)
	GF_1	84.3(3)	**94.6**(1)	87.5(2)	73.5(6)	76.4(5)	81.0(4)
$\overline{Ranking on BF_1}$		3.75	4.08	5.08	3.00	2.50	**2.08**
$\overline{Ranking on GF_1}$		4.25	4.00	3.25	3.50	3.92	**2.08**

Bold values indicate the best experimental results on each evaluation measure.

6.5 Time Consumption

Figure 8 reports the time consumption of training and testing in our AIC approach compared to three competing approaches HMM-based, Glove-best, and KR-EAR-best on two kinds of datasets of isA relationships and binary relationships (with single-word-based classes and with

Table 13. Comparison Between Our Approach and Five Baselines on 12 isA Relationships with Multi-Word-Based Classes: Part1

isA	Method	BM	PM	CM	AM	IM	AIC
* country	*BR*	**79.6**	27.4	46.0	20.3	29.7	57.9
	BP	30.8	13.0	25.1	29.0	44.4	**86.8**
	GR	24.0	54.2	65.7	85.1	89.4	**90.6**
	GP	82.4	74.8	**83.0**	78.1	81.7	66.7
	BF_1	44.4(2)	17.7(6)	32.5(4)	23.9(5)	35.6(3)	**69.5(1)**
	GF_1	37.1(6)	62.8(5)	73.4(4)	81.4(2)	**85.4(1)**	76.8(3)
* sport	*BR*	**78.7**	32.7	45.6	21.7	27.9	47.1
	BP	19.7	13.0	23.2	29.1	42.6	**61.5**
	GR	31.5	53.3	68.2	86.7	**91.5**	86.5
	GP	**87.4**	78.8	85.6	81.5	84.8	78.0
	BF_1	31.5(3)	18.6(6)	30.8(4)	24.8(5)	33.7(2)	**53.3(1)**
	GF_1	46.3(6)	63.6(5)	75.9(4)	84.0(2)	**88.0(1)**	82.1(3)
* city	*BR*	**84.5**	31.0	41.7	10.0	15.2	28.0
	BP	14.2	9.3	16.6	11.1	40.0	**63.6**
	GR	19.4	52.2	67.3	85.6	**96.3**	94.2
	GP	**88.8**	82.7	88.1	84.0	87.7	78.3
	BF_1	24.4(2)	14.3(5)	23.7(3)	10.5(6)	22.0(4)	**38.9(1)**
	GF_1	31.8(6)	64.0(5)	76.3(4)	84.8(3)	**91.8(1)**	85.5(2)
* animal	*BR*	**81.3**	37.5	61.8	11.1	17.2	18.5
	BP	10.1	7.0	17.2	7.0	**33.3**	20.5
	GR	20.5	45.5	67.4	81.0	**95.9**	1.0
	GP	90.9	86.9	**94.1**	87.6	90.8	89.4
	BF_1	17.9(4)	11.8(5)	**26.9(1)**	8.6(6)	22.7(2)	19.4(3)
	GF_1	33.5(6)	59.8(5)	78.6(4)	84.2(3)	93.3(2)	**94.4(1)**
* seasoning	*BR*	**78.1**	28.1	44.9	20.2	28.4	29.5
	BP	18.8	11.2	22.3	26.3	**41.8**	31.2
	GR	30.7	54.2	68.1	86.3	91.0	**96.8**
	GP	**87.3**	78.6	85.8	81.7	84.8	82.7
	BF_1	30.3(2)	16.0(6)	29.8(4)	22.9(5)	**33.8(1)**	30.3(2)
	GF_1	45.5(6)	64.2(5)	75.9(4)	83.9(3)	87.8(2)	**89.2(1)**
* company	*BR*	**83.5**	35.1	46.5	17.9	31.6	34.2
	BP	15.9	11.5	18.0	17.9	37.9	**54.2**
	GR	23.1	53.1	63.9	84.0	88.9	**94.3**
	GP	**89.0**	82.5	87.5	84.0	85.9	87.9
	BF_1	26.6(3)	17.3(6)	26.0(4)	17.9(5)	34.5(2)	**41.9(1)**
	GF_1	36.6(6)	64.6(5)	73.9(4)	84.0(3)	87.4(2)	**91.0(1)**

Bold values indicate the best experimental results on each evaluation measure.

multi-word-based classes) as shown in Table 6. We do not give the time consumptions in several syntactic context-based approaches (such as BM and PM) and the knowledge-based approach OLLIE here. This is because approaches of BM and PM perform worse in the assessment of sparse extraction. In the other hand, we use the online open system OLLIE as a competing approach, the time consumption is much heavier due to the remote connection to the knowledge base. It is unfair to compare each other. From Figure 8, we can see the following. First, all approaches are

Table 14. Comparison Between Our Approach and Five Baselines on 12 isA Relationships with Multi-Word-Based Classes: Part 2

isA	Method	BM	PM	CM	AM	IM	AIC
* painter	*BR*	**76.2**	37.0	44.3	21.0	29.9	27.8
	BP	20.2	14.7	24.1	29.4	**46.6**	40.0
	GR	33.1	52.5	68.9	86.8	**91.6**	88.5
	GP	**86.3**	79.0	84.7	80.7	84.2	69.7
	BF_1	31.9(3)	21.1(6)	31.3(4)	24.5(5)	**36.4(1)**	32.8(2)
	GF_1	47.9(6)	63.1(5)	76.0(4)	83.6(2)	**87.8(1)**	78.0(3)
* currency	*BR*	**79.2**	27.0	46.0	20.0	29.3	25.6
	BP	20.6	12.4	24.9	28.4	44.4	**46.9**
	GR	26.6	53.9	66.8	85.1	90.0	**98.3**
	GP	**84.2**	75.4	83.8	78.2	82.3	80.4
	BF_1	32.7(3)	17.0(6)	32.3(4)	23.5(5)	**35.3(1)**	33.2(2)
	GF_1	40.5(6)	62.8(5)	74.3(4)	81.5(3)	86.0(2)	**88.5(1)**
* disease	*BR*	**79.9**	28.3	45.8	21.1	29.6	38.5
	BP	19.2	11.1	22.9	27.3	43.0	**62.5**
	GR	27.0	50.8	66.7	85.5	90.5	**97.0**
	GP	86.1	76.5	85.1	80.7	84.1	**92.5**
	BF_1	31.0(3)	15.9(6)	30.5(4)	23.8(5)	35.1(2)	**47.6(1)**
	GF_1	41.2(6)	61.0(5)	74.8(4)	83.1(3)	87.1(2)	**94.7(1)**
* film	*BR*	**79.2**	28.0	45.4	20.9	27.8	35.4
	BP	19.5	11.8	22.8	26.2	**43.0**	30.5
	GR	25.6	52.2	64.9	83.5	90.5	**100.0**
	GP	84.4	76.1	83.9	79.1	83.0	**92.1**
	BF_1	31.3(3)	16.5(6)	30.3(4)	23.3(5)	**33.8(1)**	32.7(2)
	GF_1	39.3(6)	62.0(5)	73.2(4)	81.2(3)	86.6(2)	**95.9(1)**
* language	*BR*	**79.3**	28.2	45.4	20.6	29.1	30.9
	BP	18.7	11.0	22.2	26.3	**41.8**	32.5
	GR	27.8	52.3	66.8	85.7	90.6	**98.1**
	GP	86.5	77.7	85.5	81.3	84.6	**91.1**
	BF_1	30.2(2)	15.8(6)	29.8(4)	23.1(5)	**34.3(1)**	31.6(3)
	GF_1	42.1(6)	62.5(5)	75.0(4)	83.4(3)	87.5(2)	**94.4(1)**
* river	*BR*	**79.4**	28.6	45.6	20.6	28.7	35.2
	BP	18.8	11.2	22.4	26.3	**41.8**	31.5
	GR	29.2	53.2	67.5	85.9	90.8	**100.0**
	GP	87.3	78.3	85.8	81.6	84.7	**100.0**
	BF_1	30.4(3)	16.1(6)	30.0(4)	23.1(5)	**34.0(1)**	33.2(2)
	GF_1	43.8(6)	63.4(5)	75.5(4)	83.7(3)	87.7(2)	**100.0(1)**
$\overline{Ranking on BF_1}$		2.75	5.83	3.67	5.17	**1.75**	**1.75**
$\overline{Ranking on GF_1}$		6.00	5.00	4.00	2.75	1.67	**1.58**

Bold values indicate the best experimental results on each evaluation measure.

comparable to each other in the testing. This is because the testing time consumption in approaches of AIC, Glove-best, and KR-EAR-best mainly depends on the similarity computation. Only in the HMM-based approach, it requires to predict the test pairs using the HMM and relational *n*-gram-based models. The time overhead is a light heavier than the former three approaches.

Table 15. Comparison Between Our Approach and Three Baselines on 12 isA Relationships with Single-Word-Based Classes

Data	Method	BR	BP	GR	GP	BF1	GF1
country	HMM-based	1.0	100.0	100.0	46.6	2.0(4)	63.6(2)
	AIC	57.0	84.8	88.9	65.5	**68.2(1)**	**75.4(1)**
	Glove-best	49.1	62.7	65.1	51.7	55.1(3)	57.6(3)
	KR-EAR-best	64.2	60.7	50.3	54.0	62.4(2)	52.1(4)
sport	HMM-based	18.9	25.0	81.1	75.0	21.5(4)	77.9(2)
	AIC	62.8	42.9	84.4	92.4	50.9(2)	**88.2(1)**
	Glove-best	42.8	48.8	71.3	66.1	45.6(3)	68.6(4)
	KR-EAR-best	58.2	52.6	66.5	71.4	**55.3(1)**	68.9(3)
city	HMM-based	14.3	18.2	92.9	90.7	16.0(4)	91.8(2)
	AIC	52.4	64.7	96.2	93.9	57.9(2)	**95.0(1)**
	Glove-best	45.1	36.0	60.3	68.9	40.1(3)	64.3(4)
	KR-EAR-best	81.8	49.7	59.0	86.7	**61.8(1)**	70.2(3)
animal	HMM-based	20.8	50.0	89.4	68.9	29.4(3)	77.8(2)
	AIC	31.8	70.0	94.9	78.9	43.8(2)	**86.2(1)**
	Glove-best	26.0	33.0	76.2	69.6	29.1(4)	72.7(3)
	KR-EAR-best	75.9	47.0	61.5	85.0	**58.1(1)**	71.3(4)
seasoning	HMM-based	16.7	35.7	80.9	60.3	22.7(4)	69.1(3)
	AIC	65.4	53.1	81.3	87.8	**58.6(1)**	**84.4(1)**
	Glove-best	44.0	32.7	60.8	71.5	37.5(3)	65.7(4)
	KR-EAR-best	72.1	45.5	62.7	83.9	55.8(2)	71.8(2)
company	HMM-based	14.3	15.4	87.1	86.1	15.9(4)	86.6(2)
	AIC	20.0	50.0	98.4	93.8	28.6(2)	**96.1(1)**
	Glove-best	23.6	32.7	80.0	71.9	27.4(3)	75.7(3)
	KR-EAR-best	70.6	44.8	64.4	84.2	**54.8(1)**	73.0(4)
painter	HMM-based	50.0	11.1	63.6	93.3	18.2(3)	75.7(3)
	AIC	0.0	0.0	91.3	100.0	0.0(4)	**95.5(1)**
	Glove-best	24.3	32.0	80.0	73.2	27.6(2)	76.5(2)
	KR-EAR-best	70.2	44.0	65.5	85.1	**54.1(1)**	74.0(4)
currency	HMM-based	12.5	100.0	100.0	85.1	22.2(4)	**92.0(1)**
	AIC	83.3	26.3	77.4	98.0	40.0(2)	86.5(2)
	Glove-best	24.2	31.7	81.0	74.5	27.4(3)	77.6(3)
	KR-EAR-best	69.9	42.9	66.2	85.8	**53.2(1)**	74.7(4)
disease	HMM-based	25.0	33.3	80.0	72.7	28.6(3)	76.2(3)
	AIC	50.0	66.7	97.0	94.1	**57.1(1)**	**95.5(1)**
	Glove-best	24.1	31.7	81.6	75.2	27.4(4)	78.2(2)
	KR-EAR-best	69.6	40.2	63.5	85.5	51.0(2)	72.8(4)
film	HMM-based	51.6	53.3	69.6	68.1	54.2(2)	68.8(4)
	AIC	75.0	75.0	87.5	87.5	**75.0(1)**	**87.5(1)**
	Glove-best	23.5	31.4	81.4	74.6	26.9(4)	77.8(2)
	KR-EAR-best	67.6	39.9	63.2	84.4	50.2(3)	72.3(3)
language	HMM-based	50.0	20.0	57.1	84.2	28.6(3)	68.1(4)
	AIC	66.7	57.1	90.6	93.5	**61.5(1)**	**92.1(1)**
	Glove-best	23.2	30.6	81.4	74.9	26.4(4)	78.0(2)
	KR-EAR-best	67.2	39.5	63.6	84.6	49.8(2)	72.6(3)
river	HMM-based	100.0	11.8	34.8	100.0	21.0(3)	51.6(4)
	AIC	0.0	0.0	70.8	94.4	0.0(4)	**81.0(1)**
	Glove-best	23.3	30.3	81.4	75.4	26.4(2)	78.3(2)
	KR-EAR-best	66.9	39.1	64.0	84.8	**49.4(1)**	73.0(3)

Bold values indicate the best experimental results on each evaluation measure.

Table 16. Comparison Between Our Approach and Three Baselines on 12 isA Relationships with Multi-Word-Based Classes

Data	Method	BR	BP	GR	GP	BF1	GF1
* country	HMM-based	25.0	28.2	52.3	84.1	26.5(2)	64.5(4)
	AIC	57.9	86.8	90.6	66.7	**69.5(1)**	76.8(3)
	Glove-best	5.5	50.0	98.5	79.7	9.9(3)	**88.1(1)**
	KR-EAR-best	1.8	41.7	99.3	78.2	3.4(4)	87.5(2)
* sport	HMM-based	26.3	21.9	47.1	85.7	23.9(2)	60.8(4)
	AIC	47.1	61.5	86.5	78.0	**53.3(1)**	82.1(3)
	Glove-best	5.7	29.2	97.0	82.7	9.5(3)	**89.3(1)**
	KR-EAR-best	3.2	28.6	98.2	81.9	5.8(4)	89.3(2)
* city	HMM-based	39.5	15.2	49.7	87.7	22.0(2)	63.4(4)
	AIC	28.0	63.6	94.2	78.3	**38.9(1)**	85.5(3)
	Glove-best	5.0	50.0	99.3	88.0	9.1(3)	**93.3(1)**
	KR-EAR-best	0.0	0.0	100.0	86.2	0.0(4)	92.6(2)
* animal	HMM-based	35.9	12.6	44.9	92.1	18.7(2)	60.4(4)
	AIC	18.5	20.5	1.0	89.4	**19.4(1)**	94.4(2)
	Glove-best	2.1	33.3	99.6	91.0	3.9(3)	**95.1(1)**
	KR-EAR-best	0.0	0.0	100.0	89.4	0.0(4)	94.4(2)
* seasoning	HMM-based	23.5	21.3	47.7	86.1	20.3(2)	61.4(4)
	AIC	29.5	31.2	96.8	82.7	**30.3(1)**	89.2(2)
	Glove-best	4.5	30.6	95.5	84.0	7.9(3)	**89.4(1)**
	KR-EAR-best	3.5	28.6	98.0	81.9	6.2(4)	89.2(3)
* company	HMM-based	38.7	23.5	51.4	88.4	29.2(2)	65.0(4)
	AIC	34.2	54.2	94.3	87.9	**41.9(1)**	91.0(2)
	Glove-best	7.1	52.2	98.7	84.5	12.4(3)	**91.1(1)**
	KR-EAR-best	0.0	0.0	100.0	82.4	0.0(4)	90.3(3)
* painter	HMM-based	36.7	23.6	47.0	85.2	28.7(2)	60.6(4)
	AIC	27.8	40.0	88.5	69.7	**32.8(1)**	78.0(3)
	Glove-best	8.3	33.7	96.2	81.9	13.4(3)	88.5(2)
	KR-EAR-best	4.6	35.2	97.9	80.9	8.2(4)	88.6(1)
* currency	HMM-based	36.3	26.7	50.7	84.2	22.6(2)	63.3(4)
	AIC	25.6	46.9	98.3	80.4	**33.2(1)**	88.5(2)
	Glove-best	1.6	38.7	100.0	80.5	3.1(4)	**89.2(1)**
	KR-EAR-best	1.8	38.5	99.2	78.8	3.4(3)	87.9(3)
* disease	HMM-based	35.9	23.9	49.0	85.5	28.7(2)	62.3(4)
	AIC	38.5	62.5	97.0	92.5	**47.6(1)**	**94.7(1)**
	Glove-best	6.4	30.6	96.7	81.8	10.5(3)	88.6(3)
	KR-EAR-best	3.0	29.4	98.3	80.8	5.5(4)	88.7(2)
* film	HMM-based	30.0	24.7	48.2	84.6	27.1(2)	61.4(4)
	AIC	35.4	30.5	100.0	92.1	**32.7(1)**	**95.9(1)**
	Glove-best	6.5	29.5	96.3	81.0	10.6(3)	88.0(3)
	KR-EAR-best	1.6	29.4	99.0	79.8	3.1(4)	88.4(2)
* language	HMM-based	29.2	22.9	48.2	86.0	25.7(2)	61.8(4)
	AIC	30.9	32.5	98.1	91.1	**31.6(1)**	**94.4(1)**
	Glove-best	6.3	30.2	96.8	82.5	10.4(3)	89.1(2)
	KR-EAR-best	3.0	29.4	98.4	81.5	5.4(4)	89.1(2)
* river	HMM-based	32.7	22.6	47.5	86.1	26.7(2)	61.2(4)
	AIC	35.2	31.5	100.0	100.0	**33.2(1)**	**100.0(1)**
	Glove-best	6.1	29.2	96.8	82.6	10.2(3)	89.1(2)
	KR-EAR-best	3.5	28.6	98.0	81.7	6.3(4)	89.1(2)

Bold values indicate the best experimental results on each evaluation measure.

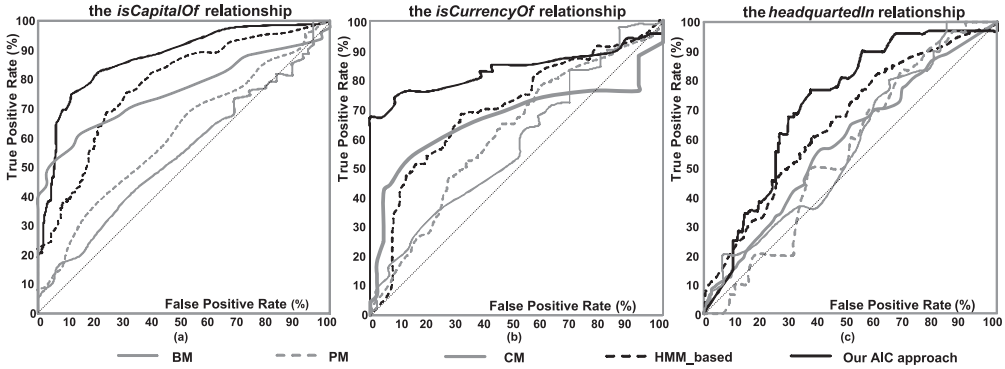

Fig. 6. ROC curves of different approaches on three binary relationships.

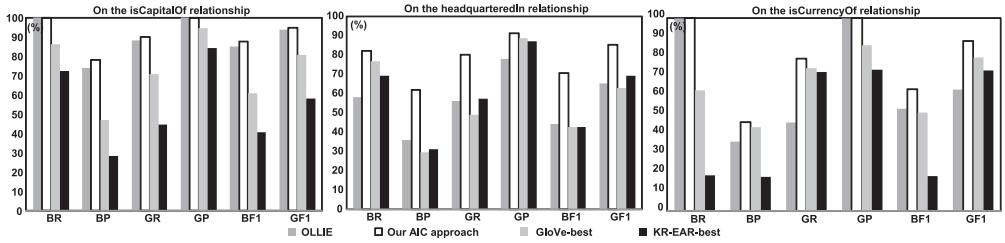

Fig. 7. Prediction results of our approach compared to approaches of OLLIE, Glove-best, and KR-EAR-best.

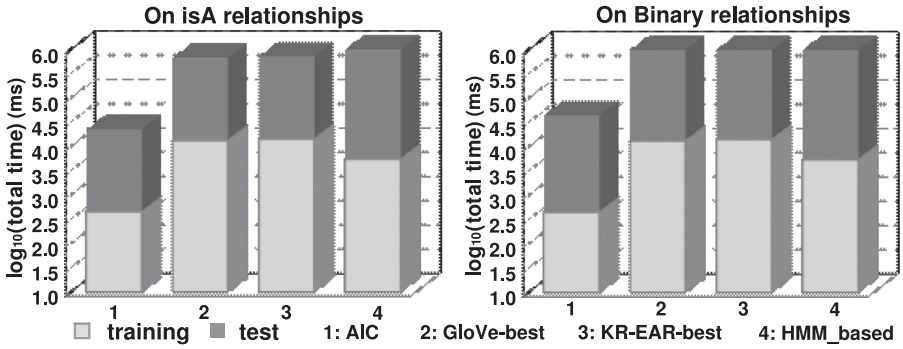

Fig. 8. Time overhead of our approach vs. baseline approaches.

Second, our AIC approach is much faster than all competing approaches in the training. The reasons are analyzed below. In our approach, the training time overhead is mainly spent on semantic context learning from the Probase semantic network in approaches of AM, IM and CM, and the generation of the classification model. Those semantic context-based approaches are faster due to the good indexing of databases in Probase, and the generation of the classification model also consumes lightly because it learns from a small set of labeled data. However, considering the HMM-based approach, the training time overhead is spent on learning the HMM and relational n-gram-based models. The time required to learn the parameters of a kth order HMM is proportional to $O(N \cdot T^{k+1})$, where N is the size of the corpus and T is the count of hidden states. This is infeasible for a large corpus. Considering the Glove-best approach, the time consumption is composed of

Table 17. Comparison Between Our Approach and Baselines on Three Binary Relationships with Multi-Word-Based Classes

Method	BR	BP	GR	GP	BF1	GF1
On isCapitalOf(* country, * city)						
BM	94.7	32.1	2.6	50.0	**48.0(1)**	4.9(8)
PM	26.3	31.3	71.8	66.7	28.6(2)	69.1(5)
CM	26.3	20.8	51.3	58.8	23.3(3)	54.8(7)
HMM_base	17.4	19.0	61.4	58.7	18.2(5)	60.0(6)
AIC	21.6	20.5	78.5	67.8	21.0 (4)	72.8(3)
Glove-best	4.3	16.7	89.1	65.1	6.9(8)	**75.2(1)**
KR-EAR-best	13.0	25.0	80.9	65.5	17.1(6)	72.4(4)
OLLIE	8.7	22.2	84.4	64.4	12.5(7)	73.1(2)
On isCurrencyOf(* country, * currency)						
BM	15.4	25.0	45.5	31.3	19.0(6)	37.0(7)
PM	38.5	55.6	63.6	46.7	**45.5(1)**	53.8(4)
CM	7.7	14.3	50.0	33.3	10.0(7)	40.0(6)
HMM_base	72.7	31.4	7.9	33.3	43.8(2)	12.8(8)
AIC	24.5	23.6	86.4	66.8	24.1(5)	75.3(2)
Glove-best	4.8	33.3	93.8	60.0	8.3(8)	73.2(3)
KR-EAR-best	17.4	66.7	94.9	66.1	27.6(3)	**77.9(1)**
OLLIE	30.4	21.9	35.9	46.7	25.5(4)	40.6(5)
On headquarteredIn(* company, * city)						
BM	92.2	45.2	10.4	62.5	**60.7(1)**	17.9(8)
PM	23.4	40.9	72.9	54.3	29.8(4)	62.2(4)
CM	37.2	39.2	53.6	51.5	38.2(2)	52.5(6)
HMM_base	30.3	29.0	43.1	44.6	29.7(5)	43.9(7)
AIC	34.5	37.2	85.3	61.3	35.8(3)	**71.4(1)**
Glove-best	6.8	35.3	90.5	56.1	11.4(8)	69.3(2)
KR-EAR-best	17.8	40.0	80.0	56.5	24.6(6)	66.2(3)
OLLIE	20.2	28.6	62.2	51.0	23.7(7)	56.1(5)

Bold values indicate the best experimental results on each evaluation measure.

constructing unigram counts, word–word co-occurrence statistics from a corpus, and training the GloVe model. Considering the KR-EAR-best approach, it is built on the supervised learning based on the knowledge graph of Freebase. The time complexities of both approaches are in direct proportion to the size of the corpus. To get the higher coverage, the selected corpus is large enough, which indicates the time consumption is costly.

7 CONCLUSIONS

We presented a lightweight, semantic context-based assessment approach for spare IEs. Our approach is build on three different semantic contexts, such as the isA-based context, attribute-based context, and class-based context. Extensive studies demonstrated that our approach outperforms several well-known approaches in the handling of IE on *F*-score. Meanwhile, it can be applied into the large scale datasets of sparse extractions due to lower time overhead. In fact, our approach can be applied in any knowledge base, if it previously provides semantic contexts mentioned in our approach. In sum, this article aims to clean the sparse extractions from Probase. We explored three semantic methods including an attribute-based method, an isA-based method, and a class-based

method. For simplicity, we aggregated these three approaches by a classical classifier logistical regression, which performs the best among seven classical classification models according to the experimental results. In our future work, we plan to explore semi-supervised classification to evaluate sparse extractions. This is because most of the sparse extractions in Probase are unlabeled and human-labeling all these pairs is very time-consuming. It is hence necessary to take advantage of unlabeled data in the classification. In this case, we will study more experiments on relationship validation using more datasets of isA relationships and binary relationships. In addition, we will try to do some exhaustive experiments by randomly select tail seeds or designing a heuristic method to perform selection for better seeds.

REFERENCES

E. Agichtein and L. Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the 5th ACM Conference on Digital Libraries (ACL'00)*. 85–94.

A. Ahuja and D. Downey. 2010. Improved extraction assessment through better language models. In *Proceedings of the Human Language Technologies (HLT'10)*. 225–228.

F. Alam, A. Corazza, A. Lavelli, and R. Zanoli. 2016. A knowledge-poor approach to chemical-disease relation extraction. *Database* 2016 (2016), 1–12.

Oznur Kirmemis Alkan and Pinar Karagoz. 2016. WaPUPS: Web access pattern extraction under user-defined pattern scoring. *Information Science* 42, 2 (2016), 261–273.

Saeed Amal, Tsvi KuFlik, and Einat Minkov. 2017. Harvesting entity-relation social networks from the web: Potential and challenges. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization (UMAP'17)*. 351–352.

Leonard E. Baum and Ted Petrie. 1966. Statistical inference for probabilistic functions of finite state Markov chains. *The Annals of Mathematical Statistics* 37, 6 (1966), 1554–1563.

Kevin Lange Di Cesare, Amal Zouaq, and Ludovic Jean-Louis. 2016. A machine learning filter for relation extraction. In *Proceedings of the 25th International Conference Companion on World Wide Web (WWW'16)*. 69–70.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics* 16, 1 (1990), 22–29.

Luciano Del Corro and Rainer Gemulla. 2013. ClausIE: Clause-based open information extraction. In *Proceedings of the 22nd International Conference Companion on World Wide Web (WWW'13)*. 355–365.

Bhavana Dalvi, William W. Cohen, and Jamie Callan. 2012. WebSets: Extracting sets of entities from the web using unsupervised information extraction. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining (WSDM'12)*. 243–252.

J. Demšar. 1961. Multiple comparisons among means. *Journal of the American Statistical Association* 56 (1961), 52–64.

J. Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* 7 (2006), 1–30.

Doug Downey, Stefan Schoenmackers, and Oren Etzioni. 2007. Sparse information extraction: Unsupervised language models to the rescue. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL'07)*. 696–703.

Doug Downeya, Oren Etzionib, and Stephen Soderland. 2010. Analysis of a probabilistic model of redundancy in unsupervised information extraction. *Artificial Intelligence* 174 (2010), 726–748.

Oren Etzioni, Michael J. Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence* 165 (2005), 91–134.

Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam. 2011. Open information extraction: The second generation. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*. 3–10.

Ronald Fagin, Benny Kimelfeld, Frederick Reiss, and Stijn Vansummeren. 2016. Declarative cleaning of inconsistencies in information extraction. *ACM Transactions on Database Systems* 41, 1 (April 2016), Article 6, 1–44.

R. Feldman and B. Rosenfeld. 2006. Boosting unsupervised relation extraction by using NER. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'06)*. 473–481.

Evgeniy Gabrilovich and Shaul Markovitch. 2009. Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research* 34 (2009), 443–498.

Kiril Gashteovski, Rainer Gemulla, and Luciano del Corro. 2017. MinIE: Minimizing facts in open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'17)*. 2630–2640.

Btihal El Ghali and Abderrahim El Qadi. 2017. Context-aware query expansion method using language models and latent semantic analyses. *Knowledge and Information Systems* 50 (2017), 751–762.

Dihong Gong, Daisy Zhe Wang, and Yang Peng. 2017. Multimodal learning for web information extraction. In *Proceedings of the Multimedia Conference (MM'17)*. 288–296.

Pankaj Gulhane, Rajeev Rastogi, Srinivasan H Sengamedu, and Ashwin Tengli. 2010. Exploiting content redundancy for web information extraction. In *Proceedings of the World Wide Web (WWW'10)*. 1105–1106.

Maeda F. Hanafi, Azza Abouzied, Laura Chiticariu, and Yunyao Li. 2017. Synthesizing extraction rules from user examples with SEER. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD'17*). 1687–1690.

Z. Harris. 1985. *Distributional Structure*. The Philosophy of Linguistics. New York: Oxford University Press.

M. A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the International Conference on Computational Linguistics (COLING'92)*. 539–545.

Johannes Hoffart, Stephan Seufert, Dat Ba Nguyen, Martin Theobald, and Gerhard Weikum. 2012. KORE: Keyphrase overlap relatedness for entity disambiguation. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM'12)*. 545–554.

Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, Edwin L. Kelham, Gerard de Melo, and Gerhard Weikum. 2011. YAGO2: Exploring and querying world knowledge in time, space, context, and many languages. In *Proceedings of the 20th International Conference Companion on World Wide Web (WWW'11*). 229–232.

Yuzhe Jin, Emre Kiciman, Kuansan Wang, and Ricky Loynd. 2014. Entity linking at the tail: Sparse signals, unknown entities, and phrase models. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining (WSDM'14)*. 453–462.

Mayank Kejriwal and Pedro Szekely. 2017. Information extraction in illicit domains. In *Proceedings of the 26th International Conference on World Wide Web (WWW'17)*. 997–1006.

Dongwoo Kim, Haixun Wang, and Alice H. Oh. 2013. Context-dependent conceptualization. In *Proceedings of the Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*. 2654–2661.

Sarath Kumar Kondreddi, Peter Triantafillou, and Gerhard Weikum. 2014. Combining information extraction and human computing for crowdsourced knowledge acquisition. In *Proceedings of the 30th International Conference on Data Engineering (ICDE'14)*. 988–999.

Taesung Lee, Zhongyuan Wang, Haixun Wang, and Seung won Hwang. 2011. Web scale taxonomy cleansing. In *Proceedings of the 37th International Conference on Very Large Data Bases (VLDB'11)*. 1295–1306.

Taesung Lee, Zhongyuan Wang, Haixun Wang, and Seung won Hwang. 2013. Attribute extraction and scoring: A probabilistic approach. In *Proceedings of the 29th International Conference on Data Engineering (ICDE'13)*. 194–205.

Jiewu Leng and Pingyu Jiang. 2016. A deep learning approach for relationship extraction from interaction context in social manufacturing paradigm. *Knowledge-Based Systems* 100 (2016), 188–199.

Peipei Li, Haixun Wang, Kenny Zhu, Zhongyuan Wang, and Xindong Wu. 2015. A large probabilistic semantic network based approach to compute term similarity. *IEEE Transactions on Knowledge and Data Engineering* 27 (2015), 2604–2617.

Yang Li, Shulong Tan, Huan Sun, Jiawei Han, Dan Roth, and Xifeng Yan. 2016. Entity disambiguation with linkless knowledge bases. In *Proceedings of the 25th International Conference on World Wide Web (WWW'16)*. 1261–1270.

Rinaldo Lima, Bernard Espinasse, and Fred Freitas. 2017. OntoILPER: An ontology- and inductive logic programming-based system to extract entities and relations from text. *Knowledge and Information Systems* (2017), 1–33. DOI: http://dx.doi.org/10.1007/s1011

Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2016. Knowledge representation learning with entities, attributes and relations. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI'16)*. 2866–2872.

Kevin Meijer, Flavius Frasincar, and Frederik Hogenboom. 2014. A semantic approach for extracting domain taxonomies from text. *Decision Support Systems* 62 (2014), 78–93.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv:1301.3781*.

Masayuki Okamoto, Zifei Shan, and Ryohei Orihara. 2017. Applying information extraction for patent structure analysis. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'17)*. 987–992.

Sergio Oramas, Mohamed Sordo, and Luis Espinosa-Anke. 2015. A rule-based approach to extracting relations from music tidbits. In *Proceedings of the 24th International Conference on World Wide Web (WWW'15)*. 661–666.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*. 1532–1543.

Simone P. Ponzetto and Michael Strube. 2007. Deriving a large-scale taxonomy from wikipedia. In *Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI'07)*. 1440–1445.

Priya Radhakrishnan and Vasudeva Varma. 2013. Extracting semantic knowledge from wikipedia category names. In *Proceedings of the Workshop on Automated Knowledge Base Construction (CIKM'13)*. 109–114.

Alexander J. Ratner, Stephen H. Bach, Henry R. Ehrenberg, and Chris RÍę. 2017. Snorkel: Fast training set generation for information extraction. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD'17)*. 1683–1686.

Ridho Reinanda, Edgar Meij, and Maarten de Ri-jke. 2016. Document filtering for long-tail entities. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM'16)*. 771–780.

Alan Ritter, Stephen Soderland, and Oren Etzioni. 2009. What is this, anyway: Automatic hypernym discovery. In *Proceedings of the AAAI Spring Symposium on Learning by Reading and Learning to Read*. 88–93.

Michael Schmitz, Robert Bart, Stephen Soderland, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL'12)*. 523–534.

Mondher Sendi and Mohamed Nazih Omri. 2015. Biomedical concept extraction based information retrieval model: Application on the MeSH. In *Proceedings of the 15th International Conference on Intelligent Systems Design and Applications (ISDA'15)*. 40–45.

Yangqiu Song, Haixun Wang, Zhongyuan Wang, Hongsong Li, and Weizhu Chen. 2011. Short text conceptualization using a probabilistic knowledgebase. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*. 2330–2336.

John A. Swets. 1996. *Signal Detection Theory and ROC Analysis in Psychology and Diagnostics: Collected Papers*. Lawrence Erlbaum Associates, Mahwah, NJ.

Bilyana Taneva and Gerhard Weikum. 2013. Gem-based entity-knowledge maintenance. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM'13)*. 149–158.

Fei Wu, Raphael Hoffmann, and Daniel S. Weld. 2008. Information extraction from wikipedia: Moving down the long tail. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'08)*. 731–739.

F. Wu and D. S. Weld. 2010. Open information extraction using wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL'10)*. 118–127.

Wentao Wu, Hongsong Li, Haixun Wang, and Kenny Q. Zhu. 2012. Probase: A probabilistic taxonomy for text understanding. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD'12)*. 481–492.

Peng Yan and Wei Jin. 2017. Building semantic kernels for cross-document knowledge discovery using Wikipedia. *Knowledge and Information Systems* 51 (2017), 287–310.

Alexander Yates, Michael Cafarella, Michele Banko, Oren Etzioni, Matthew Broadhead, and Stephen Soderland. 2007. TextRunner: Open information extraction on the web. In *Proceedings of the Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations (NAACL HLT Demonstration Program'07)*. 25–26.

Jingyuan Zhang, Roger Jie Luo, Altaf Rahman, Yi Chang, and Philip S. Yu. 2015. Learning entity types from query logs via graph-based modeling. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM'15)*. 603–702.

Minling Zhang and Lei Wu. 2015. LIFT: Multi-label learning with label-specific features. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37 (2015), 107–120.

Guodong Zhou, Longhua Qian, and Jianxi Fan. 2010. Tree kernel-based semantic relation extraction with rich syntactic and semantic information. *Information Sciences* 180 (2010), 1313–1325.

Jun Zhu, Zaiqing Nie, Xiaojiang Liu, Bo Zhang, and Ji rong Wen. 2009. StatSnowball: A statistical approach to extracting entity relationships. In *Proceedings of the 18th International Conference on World Wide Web (WWW'09)*. 101–110.

Received September 2017; revised February 2018; accepted March 2018